

**Universidad ORT Uruguay
Facultad de Ingeniería**

**Aplicación Móvil de Apoyo para C.U.T.C.S.A.:
Compañía Uruguaya de Transportes
Colectivos Sociedad Anónima**

Entregado como requisito para la obtención del título de
Ingeniero en Sistemas

Andrés Amegeiras 123768

Diego Barcia 173809

Juan Pedro Trecca 177244

Tutor: Amalia Álvarez

2016

Declaración de autoría

Nosotros, Andrés Amegeiras, Diego Barcia y Juan Pedro Trecca, declaramos que el trabajo que se presenta en esa obra es de nuestra propia mano. Podemos asegurar que:

- La obra fue producida en su totalidad mientras realizábamos Aplicación Móvil de Apoyo para C.U.T.C.S.A.;
- Cuando hemos consultado el trabajo publicado por otros, lo hemos atribuido con claridad;
- Cuando hemos citado obras de otros, hemos indicado las fuentes. Con excepción de estas citas, la obra es enteramente nuestra;
- En la obra, hemos acusado recibo de las ayudas recibidas;
- Cuando la obra se basa en trabajo realizado conjuntamente con otros, hemos explicado claramente qué fue contribuido por otros, y qué fue contribuido por nosotros;
- Ninguna parte de este trabajo ha sido publicada previamente a su entrega, excepto donde se han realizado las aclaraciones correspondientes.



Andrés Amegeiras
3 de marzo de 2016



Diego Barcia
3 de marzo de 2016



Juan Pedro Trecca
3 de marzo de 2016

Agradecimientos

En primer lugar, a Amalia Álvarez por la dedicación y apoyo que nos dio a lo largo de este año. Estando siempre disponible para ayudarnos y guiarnos en la toma de decisiones a lo largo del proyecto.

A la empresa C.U.T.C.S.A. por brindarnos la oportunidad de realizar un proyecto aplicado a la vida real, y por todo el apoyo recibido para que el mismo se lleve a cabo de la mejor manera.

Manuel Ares, Gerente de Área de Sistemas y Estudios Tecnológicos de la empresa C.U.C.T.S.A., quien ofició de *Sponsor* del equipo ante la empresa, facilitándonos la comunicación con los empleados de la misma. Además de haber tenido varias reuniones con el equipo.

Federico Rena, administrador de redes de la empresa C.U.T.C.S.A. por sus grandes aportes en lo referido a seguridad y pruebas en ambiente de producción.

A Verónica Barcia por la ayuda ofrecida en el diseño de varias de las imágenes y logos utilizados en la aplicación.

Al equipo de ORTs por su apoyo en la realización de planes y respuesta a dudas referidas a las distintas áreas de tutoría.

A los revisores que tuvimos a lo largo del año, Gastón Mousques, Martin Solari y Marcelo Cagnani, por los consejos y apreciaciones recibidas en estas instancias, que nos ayudaron a mejorar el enfoque del proyecto y la forma de presentar el mismo a terceros.

A Daniel Mordeki por sus invaluable aportes para las correcciones y mejora de la interfaz de usuario y usabilidad de la aplicación.

A nuestras familias y amigos por el apoyo que nos ofrecieron durante el transcurso del año de desarrollo para el proyecto.

Abstract

El objetivo del proyecto es crear una aplicación móvil que ayude y apoye la gestión, comunicación y trabajo que realizan los distintos funcionarios en su día a día, en todas las áreas de la empresa de C.U.T.C.S.A.

El resultado del proyecto permitirá dar soporte, agilizar y mejorar la organización de las tareas, aumentar la eficiencia y eficacia del funcionario y mejorar o complementar las comunicaciones entre empleados, departamentos y áreas.

El funcionario podrá tener de forma más accesible, rápida y sencilla la información que utiliza durante su jornada laboral, aprovechando las ventajas que ofrecen las características de un dispositivo móvil.

Dado el tamaño de la empresa, compuesta por cinco áreas operativas, 23 departamentos y alrededor de 5000 funcionarios, el producto final no completa todas las funcionalidades requeridas para la empresa. Por dicha razón, el proyecto de fin de carrera es la etapa inicial de un proyecto que se continuará luego por la empresa.

El proyecto se divide en tres etapas:

- La primera se profundiza en la descripción del producto, análisis del problema y planificación de su desarrollo.
- La segunda incluye las evoluciones que permiten el desarrollo de varios módulos del producto (los módulos consisten en conjuntos de requerimientos que forman parte de la aplicación móvil organizados por Área o Departamento de la empresa), que incluyen a Inspectores, Largadores, Conductores, Guardas y Conductores Cobradores, roles pertenecientes a la empresa.
- Una etapa final para realizar la documentación requerida por la Facultad, y así concluir con el proyecto.

El producto está compuesto por una aplicación móvil organizada en módulos, cada uno ofreciendo funcionalidad para cada Área o Sector de la empresa; y por un *backend* que cuenta con los Servicios web que consume la aplicación móvil, así como el desarrollo y gestión de la base de datos que ofrece la información necesaria.

Palabras clave

C.U.T.C.S.A.; Transporte Colectivo; Dispositivos móviles; Android; Control Boletos; Gestión Transporte; Proyecto Modular; Tradicional Evolutivo;

Índice

Declaración de autoría	2
Agradecimientos.....	3
Abstract	4
Palabras clave	5
Índice	6
Glosario.....	13
1. Introducción	17
1.1. Entorno Conceptual de Software Factory	17
1.2. Objetivos principales.....	17
1.2.1. Objetivos del proyecto	18
1.2.2. Objetivos Académicos	18
1.3. Descripción del cliente.....	18
1.4. Descripción del equipo de trabajo.....	19
1.5. Organización del documento.....	20
2. Planteamiento del problema.....	22
2.1. Contexto del problema	22
2.1.1. Problemática general.....	22
2.1.2. Problemas abordados en el proyecto.	23
2.2. Partes interesadas del proyecto	25
2.3. Principales aspectos de la solución propuesta	26
2.4. Supuestos y restricciones	28
2.4.1. Supuestos	28
2.4.2. Restricciones.....	29
3. Proceso y ciclo de vida.....	30
3.1. Ciclo de vida	30
3.2. Metodologías de referencia.....	32
4. Ingeniería de requerimientos	33
4.1. Introducción.....	33
4.2. Proceso de Ingeniería de Requerimientos.....	33
4.2.1. Relevamiento.....	34
4.2.2. Análisis	36
4.2.3. Especificación	36
4.2.4. Validación	37

4.3.	Estructura utilizada para la especificación	37
4.4.	Criterios de la priorización de los requerimientos	38
4.5.	Requerimientos funcionales	40
4.6.	Requerimientos no funcionales	42
4.6.1.	Características de calidad	42
4.6.2.	Restricciones.....	44
4.7.	Descripción de funcionalidad principal.....	44
4.8.	Conclusiones	46
5.	Arquitectura.....	47
5.1.	Descripción General.....	48
5.2.	Arquitectura vs Atributos de calidad	49
5.2.1.	Fiabilidad.....	50
5.2.2.	Usabilidad	50
5.2.3.	Eficiencia.....	51
5.2.4.	Mantenibilidad	55
5.2.5.	Seguridad	55
5.3.	Descripción de la arquitectura.....	58
5.3.1.	Patrones.....	58
5.3.2.	Vistas Arquitectónicas	58
5.4.	Decisiones generales de tecnologías utilizadas.....	67
5.4.1.	Herramientas de desarrollo.....	68
5.5.	Conclusiones	69
5.5.1.	Lecciones Generales	69
5.5.2.	Lecciones Sobre tecnologías.....	69
6.	Proceso de desarrollo	71
6.1.	Características del equipo.....	71
6.2.	Responsabilidades	71
6.3.	Estrategia de desarrollo	72
6.4.	Alcance desarrollado.....	74
6.5.	Conclusiones	75
7.	Gestión del proyecto	77
7.1.	Estrategias de gestión y planificación temporal de tareas.....	77
7.2.	Gestión de las comunicaciones.....	79
7.2.1.	Comunicaciones dentro del equipo.....	79
7.2.2.	Partes interesadas del proyecto.....	79

7.2.3.	Resumen de comunicaciones	80
7.3.	Planificación de tareas y recursos.....	81
7.4.	Herramientas de seguimiento de tareas	85
7.5.	Estimaciones	87
7.6.	Seguimiento y evaluación de actividades.....	87
7.7.	Gestión de riesgos.....	89
7.8.	Métricas	93
7.8.1.	Métricas generales	93
7.8.2.	Métricas de la etapa de desarrollo.....	100
7.9.	Principales decisiones de gestión	103
7.10.	Conclusiones y lecciones aprendidas	104
8.	Aseguramiento de calidad	106
8.1.	Objetivos de la calidad	106
8.2.	Aseguramiento de la calidad	106
8.2.1.	Planificación de la Calidad	106
8.2.2.	Estándares	107
8.2.3.	Revisiones	108
8.2.4.	Validaciones.....	110
8.2.5.	Revisiones de documentos y actividades	111
8.3.	Usabilidad	113
8.3.1.	Trabajo de interfaz de usuario	113
8.3.2.	Aseguramiento de Usabilidad	115
8.4.	<i>Testing</i>	115
8.4.1.	Planificación de la prueba	115
8.4.2.	Ejecución y resultados	120
8.5.	Herramientas	122
8.6.	Métricas de Producto	122
8.7.	Conclusiones y lecciones aprendidas.....	126
9.	Gestión de la configuración	127
9.1.	Identificación de los elementos de la configuración	127
9.2.	Selección de herramientas.....	127
9.3.	Gestión de repositorio	129
9.4.	<i>Release</i> de versiones.....	130
9.5.	Conclusiones	131
10.	Conclusiones.....	132

10.1.	Conclusiones generales	132
10.2.	Líneas de trabajo para el futuro.....	135
11.	Bibliografía.....	136
12.	Anexos	142
12.1.	Anexo 1 - Situación actual de la empresa	142
12.1.1.	Situación actual de la empresa	142
12.1.2.	Visión del Producto	157
12.1.3.	Anexos.....	161
12.2.	Anexo 2 – ESRE	167
12.2.1.	Introducción.....	167
12.2.2.	Descripción General.....	167
12.2.3.	Requerimientos.....	169
12.2.4.	Casos de Uso	174
12.3.	Anexo 3 - Encuesta Inspectores	213
12.4.	Anexo 4 - Documentación de Arquitectura	223
12.4.1.	Introducción.....	223
12.4.2.	Propósito.....	223
12.4.3.	Comportamiento del Sistema	223
12.4.4.	Patrones y Principios de Diseño.....	223
12.4.5.	Principios.....	224
12.4.5.2.	Principios a nivel de clase.....	225
12.4.6.	Patrones	226
12.4.7.	Tecnología a utilizar	228
12.4.8.	Diagramas	228
12.4.8.1.	Capas Lógicas.....	228
12.4.8.2.	Diagrama de Deployment	230
12.4.8.3.	Diagrama de Componentes.....	233
12.4.8.4.	Diagramas de Paquetes.....	235
12.4.8.4.1.	Paquetes de la aplicación Móvil.....	235
12.4.8.5.	Diagramas de Clases.....	238
12.4.8.5.1.	Diagrama de clases de la aplicación móvil.....	238
12.4.8.5.2.	Diagrama de clases del servidor.....	240
12.4.8.5.3.	Diagrama de clases del intermediario.....	241
12.4.8.5.4.	Diagramas de Secuencia.....	242
12.4.9.	Descripción de la Persistencia del sistema	242

12.4.10.	Cumplimientos con los Atributos de Calidad.....	243
12.4.10.1.	Fiabilidad	243
12.4.10.2.	Usabilidad.....	243
12.4.10.3.	Eficiencia.....	244
12.4.10.4.	Mantenibilidad	246
12.4.10.5.	Seguridad.....	247
12.4.11.	Avances de Investigación	248
12.5.	Anexo 5 - Plan de Proyecto	252
12.5.1.	Resumen ejecutivo.....	252
12.5.2.	Objetivos del Producto	252
12.5.3.	Gestión del alcance	252
12.5.4.	Alcance del producto	253
12.5.5.	Estimaciones	254
12.5.6.	Cronograma de trabajo.....	255
12.5.7.	Análisis de la evolución del cronograma	258
12.5.8.	Gestión de riesgos.....	259
12.5.9.	Gestión del Esfuerzo	265
12.5.10.	Organización de partes interesadas	269
12.5.11.	Dependencias Relevantes.....	275
12.5.12.	Gestión de comunicaciones.....	276
12.5.13.	Supuestos, restricciones y obligaciones	279
12.5.14.	Anexos.....	284
12.6.	Anexo 6 - Plan de Calidad.....	292
12.6.1.	Metodología de trabajo	293
12.6.2.	Plan de la calidad	293
12.7.	Anexo 7 - Plan de SQA.....	297
12.7.1.	Propósito.....	298
12.7.2.	Referencias.....	298
12.7.3.	Definiciones	298
12.7.4.	Gestión	298
12.7.5.	Estándares, prácticas y convenciones	301
12.7.6.	Revisiones	302
12.7.7.	Validaciones	303
12.7.8.	Pruebas	303
12.7.9.	Reporte de problemas y acciones correctivas.....	305

12.7.10.	Herramientas, técnicas y metodologías	307
12.7.11.	Control de código y de medios	308
12.7.12.	Formación	308
12.7.13.	Anexos.....	309
12.8.	Anexo 8 - Informe Primera Revisión	320
12.8.1.	Fortalezas del Grupo	320
12.8.2.	Oportunidades de Mejora	320
12.8.3.	Acciones a Realizar.....	321
12.9.	Anexo 9 - Informe Segunda Revisión	322
12.9.1.	Oportunidades de Mejora	322
12.9.2.	Acciones a Realizar.....	323
12.10.	Anexo 10 - Informe Tercera Revisión	324
12.10.1.	Fortalezas del Grupo	324
12.10.2.	Oportunidades de Mejora	324
12.10.3.	Acciones a Realizar.....	325
12.11.	Anexo 11 - Evolución de la Interfaz de Usuario	326
12.11.1.	Resumen ejecutivo.....	326
12.11.2.	Objetivo.....	326
12.11.3.	Reunión	326
12.11.4.	Las 3 reglas de oro	331
12.11.5.	Correcciones y evolución	331
12.12.	Anexo 12 - Análisis de Heurísticas de Nielsen.....	362
12.13.	Anexo 13 - Análisis de problemas de visión cromática.....	365
12.13.1.	Situación.....	365
12.13.2.	Solución.....	365
12.13.3.	Bibliografía	365
12.13.4.	Anexo	366
12.14.	Anexo 14 - Estudio Facilidad Aprendizaje	369
12.15.	Anexo 15 - Prueba de carga y eficiencia	372
12.16.	Anexo 16 - Plan SCM	375
12.16.1.	Objetivos	375
12.16.2.	Definiciones	375
12.16.3.	Responsabilidades	375
12.16.4.	Identificación de los elementos de la configuración	376
12.16.5.	Selección de herramientas.....	376

12.16.6.	Gestión del repositorio	379
12.16.7.	<i>Release</i> de versiones.....	381
12.16.8.	Respaldos y actualizaciones.....	382
12.17.	Anexo 17 - Pruebas de seguridad.....	383
12.18.	Anexo 18 - Aplicación Manage It.....	386
12.19.	Anexo 19 - Registro de horas	387
12.20.	Anexo 20 - ToDoList	388
12.21.	Anexo 21 – Reunión final con el Sub Gerente General de C.U.T.C.S.A.	389
12.22.	Anexo 22 – Análisis de puntos de función	391
12.22.1.	Primera evolución	391
12.22.2.	Segunda evolución	394
12.22.3.	Tercera evolución.....	400
12.23.	Anexo 23 – Reunión final con <i>Sponsor</i>	405

Glosario

.NET: Framework de Microsoft para el rápido desarrollo de aplicaciones.

ANDROID: Sistema operativo diseñado para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas.

ACTIVITY: Del término obtenido de Android, *“es un componente de la aplicación móvil, que provee una pantalla en la que los usuarios pueden interactuar para realizar una acción, tales como llamadas telefónicas, tomar una fotografía, enviar un correo electrónico, o ver un mapa.”* [1]

AES: *Advanced Encryption Standard* – en español, estándar de encriptado avanzado–, *“es un esquema de cifrados por bloques adoptado como un estándar de cifrado.”* [2]

API: *Application Programming Interface* – en español, interfaz de programación de aplicaciones –, *“es un conjunto de rutinas, protocolos y herramientas para construir programas y aplicaciones.”* [3]

BACKEND: Sistema relacionado al usuario administrador de una aplicación que permite la administración de la misma.

C# (C SHARP): Lenguaje de programación con soporte para múltiples paradigmas desarrollado por Microsoft.

CALIDAD: Es el grado en el que un conjunto de características inherentes cumple con los requisitos.

CÓDIGO QR: Imagen para almacenar información a partir de una matriz de puntos. [4]

C.U.T.C.S.A.: Compañía Uruguaya de Transportes Colectivos Sociedad Anónima, nombre de la empresa a quien se le realizó el proyecto.

DBA: Es la persona responsable de los aspectos ambientales de una base de datos. En general esto incluye crear y probar Respaldos, así como verificar o ayudar a la verificación en la integridad de datos

ENTITY FRAMEWORK: Conjunto de tecnologías que permiten el desarrollo de software orientado a datos.

ESRE: Especificación de Requerimientos.

FRAMEWORK: Estructura conceptual y tecnológica, con módulos de software concretos, con base a la cual otro proyecto de software puede ser más fácilmente organizado y desarrollado.

FRONTEND: Sistema relacionado al usuario que utilizará el sistema, en el caso del proyecto, una aplicación móvil.

GIF: *Graphics Interchange Format* – en español, formato de intercambio de gráficos – “es un formato gráfico usado tanto para imágenes como para animaciones.” [5]

GOOGLE PLACES: API ofrecida por Google para para obtener información de lugares dentro del mapa del mundo. [6]

IIS: *Internet Information Services*, es un servidor web y un conjunto de servicios para el sistema operativo Microsoft Windows.

I.M: Intendencia de Montevideo, quien realiza parte de la gestión de datos relativos al tránsito. [7]

JSON: *JavaScript Object Notation*, “es un formato ligero de intercambio de datos.” [8]

LAYOUT: Del término obtenido de Android, “es un componente que define la estructura visual para la interfaz de usuario.” [9]

LÍNEA: Agrupación de recorridos, que tienen un origen y destino similar.

MÁQUINA EXPENDEDORA DE BOLETOS: Elemento que se encuentra sobre el ómnibus, utilizado para realizar la venta de boletos a los pasajeros, e impresión del reporte de inspección para controlar el servicio.

NFC: *Near Field Communication* – en español, comunicación de campo cercano –, “es un conjunto de protocolos de comunicación, permitiendo el establecimiento de una comunicación entre dos dispositivos electrónicos.” [10]

ORTsf: *ORT Software Factory*, “organización académica dedicada a la enseñanza de prácticas de Ingeniería de Software, a la mejora de procesos de software, a la transferencia de tecnología a la industria y a la producción de software.” [11]

PF: Punto de Función, “*es un elemento del método utilizado en ingeniería del software para medir el tamaño del software.*” [12]

RECORRIDO: Agrupación de variantes.

REFACTORIZAR: Dejar de forma más prolija el código realizado, respetando normas y estándares de codificación.

RNF: Requerimientos no funcionales, especifican criterios que pueden usarse para juzgar la operación de un sistema en lugar de sus comportamientos específicos.

RETRABAJO: Actividad que consiste en rehacer o corregir defectos de un producto.

SCM: *Software Configuration Management*, tareas referentes a la gestión de configuración del software.

Servicio: también denominado “cartón”, es un conjunto de variantes que debe realizar un coche en una jornada laboral.

SHA-1 y SHA2: *Secure Hash Algorithm 1 y 2*– en español, algoritmo de *hash* seguro 1 y 2– es una función criptográfica *hash*, SHA2 publicado posteriormente al SHA1, usado para la transmisión segura de datos. [13]

SQA: *Software Quality Assurance*, tareas referentes al aseguramiento de la calidad del software.

TI: Tecnología de Información, “*es una aplicación de computadoras y equipos de telecomunicación para almacenar, obtener, transmitir y manipular datos*”. [14]

TSL/SSL: *Transport Layer Security/ Secure Sockets Layer* – en español, seguridad de la capa de transporte y capa de conexión segura – son protocolos de seguridad basados criptografía de claves públicas. [15]

UI: *User Interface* – en español, interfaz de usuario – es el diseño realizado para la interacción entre un usuario y una computadora. [16]

UML: *Unified Modeling Language* – en español, lenguaje unificado de modelado –, es un lenguaje “*utilizado para especificar, visualizar y documentar modelos de sistemas de software, introduciendo su estructura y diseño.*” [17]

VARIANTE: Ruta que tiene un origen y destino único, del cual los coches asignados a ella deben respetar para transportar a los pasajeros.

VIAJE: Asignación de una variante a un coche para un día y hora determinado.

1. Introducción

El presente documento describe el proyecto de fin de carrera denominado “Aplicación Móvil de Apoyo para C.U.T.C.S.A.”, realizado para cumplir el requisito de la carrera Ingeniería en Sistemas, por intermedio del Laboratorio de Ingeniería de Software de la Universidad ORT Uruguay.

El proyecto estuvo comprendido entre los meses de marzo de 2015 y marzo de 2016. Fue realizado para la empresa de transporte colectivo urbano C.U.T.C.S.A. En dicho proyecto participaron Andrés Amegeiras, Diego Barcia, Juan Pedro Trecca. La tutoría del proyecto estuvo a cargo de la Lic. Amalia Álvarez.

1.1. Entorno Conceptual de Software Factory

El Laboratorio denominado ORT Software Factory (ORTsf), es una organización académica dedicada a la enseñanza de prácticas de Ingeniería de Software, a la mejora de procesos de software, a la transferencia de tecnología a la industria y a la producción de software [18].

ORTsf está abocada fundamentalmente a desarrollar en los alumnos las habilidades que un profesional de las Tecnologías de la Información debe dominar y aplicar. Esto se logra a través de un método de enseñanza diseñado para que estudiantes de fin de carrera, apoyados por tutores especializados, trabajen en equipos con proyectos reales y aplicando prácticas avanzadas de Ingeniería de Software.

Los objetivos académicos de ORTsf son:

- Formar en la producción regida por un Proceso de Software y en técnicas, herramientas y métodos de Ingeniería de Software.
- Promover el trabajo en equipo.
- Transferencia de tecnología a la industria [11].

1.2. Objetivos principales

En la siguiente sección se describen los objetivos más importantes de la tesis de grado, que surgieron a partir de los objetivos personales de los integrantes que se transformaron en objetivos comunes al grupo. Estos se dividen en objetivos del proyecto y objetivos académicos.

1.2.1. Objetivos del proyecto

- El objetivo del producto será el ayudar a mejorar procesos de comunicación y manejo de información para diferentes áreas de la empresa C.U.T.C.S.A., así como también mejorar la eficiencia y eficacia de tareas mediante la digitalización de información y procesos.
- El objetivo del proyecto es crear una aplicación Android que sirva para mejorar y facilitar el proceso de trabajo de al menos 2 roles de empleados distintos dentro de la empresa.
- Crear una solución tecnológica sostenible, que facilite su análisis posterior y extensibilidad a nuevos procesos del negocio y actores dentro de la organización, basándose en la premisa que la aplicación seguirá siendo desarrollada en un futuro por el cliente.
- Lograr cumplir los compromisos asumidos con el cliente y éste quede satisfecho con el producto final.
- Aportar innovación al sistema de trabajo de la empresa.

1.2.2. Objetivos Académicos

- Utilizar las habilidades técnicas adquiridas en la carrera para poder llevar a cabo este proyecto tecnológico.
- Obtener experiencia real en gestión de proyectos y en nuevas tecnologías para generar una base de lecciones aprendidas y, de esta forma, buscar crecer a un nivel profesional.
- Aprender a gestionar el proyecto en todas sus fases y desafíos. Debido a que el proyecto tendrá ciclos completos de desarrollo, se deberán poder gestionar todas las fases de la ingeniería de software para poder lograr un producto de calidad para el cliente.
- Lograr la aprobación del proyecto.

1.3. Descripción del cliente

C.U.T.C.S.A. es la empresa de transporte más grande de Uruguay, en cuanto a cantidad de unidades y cobertura de servicios. Brinda el servicio de transporte urbano en Montevideo desde 1937 y actualmente presta servicio suburbano llegando a Canelones y San José, entre otros servicios.

Principalmente, la empresa brinda servicio de transporte y una serie de actividades comerciales e industriales que constituyen la infraestructura necesaria para posibilitar la mayor eficiencia del servicio principal, logrando objetivos de seguridad de abastecimiento y abaratamiento de los costos.

Estas actividades son: almacenes de repuestos, surtidores de combustibles, reparaciones de carrocerías, recauchutajes de cubiertas, talleres mecánicos y de electricidad, servicio de gomería, auxilio y remolques, estacionamiento de vehículos, servicio de engrases, etc.

C.U.T.C.S.A. es una organización socio-económica en la que se concretan los principios de democracia participativa y solidaria.

Está integrada por alrededor de 1200 conductores, 1100 conductores cobradores, 1200 guardas, 300 controladores de servicios, 300 que mantienen la flota, y 600 directores, administradores y de servicios generales; así sumando casi 5000 trabajadores [19].

Con 122 recorridos y una flota de 1090 coches, a diario se trasladan 650.000 personas en 11.100 viajes, recorriendo 280.000 kilómetros.

1.4. Descripción del equipo de trabajo

El equipo se encuentra integrado por 3 estudiantes de la carrera de Ingeniería en Sistemas. Como se plantea previamente, uno de los objetivos es poder plasmar los conocimientos adquiridos durante la carrera, por lo que las responsabilidades de cada uno se distribuyeron en función de sus conocimientos y afinidad a las tareas involucradas.

En el momento de distribución de tareas y roles dentro del equipo, se plantearon ciertas reglas tanto para los roles en sí y para la asignación de los mismos:

- Todos los integrantes del equipo cumplirán con el rol de Desarrollador.
- Todos los integrantes del equipo cumplirán con el rol de *Tester*.
- Cada área de proceso tendrá al menos 2 responsables, donde uno será el responsable del área y otro quien lo pueda complementar.

A continuación, se detalla la distribución de roles dentro del proyecto.

Andrés Amegeiras

- Área de responsabilidad principal: Ingeniería en Requerimientos, Gestión de la configuración.
- Área de responsabilidad secundaria: Gestión de Calidad, Gestión del proyecto

Diego Barcia

- Área de responsabilidad principal: Gestión del proyecto
- Área de responsabilidad secundaria: Ingeniería en Requerimientos, Arquitectura

Juan Pedro Trecca

- Área de responsabilidad principal: Gestión de Calidad y Arquitectura
- Área de responsabilidad secundaria: SCM

Además de los roles que se mencionan, según las prioridades correspondientes, durante el transcurso del proyecto el equipo colabora en las tareas de cada uno de los procesos donde desempeña un rol de apoyo. Por esto, las tareas que ejecuta cada uno de los integrantes no quedan puramente ligadas a los roles que se mencionan previamente.

1.5. Organización del documento

A continuación, se detallan los capítulos que se desarrollan en el presente documento.

Planteamiento del problema: En este capítulo se plantea el problema que se desea resolver, identificando el contexto y sus usuarios. Se detallan las funcionalidades y sus restricciones

Proceso y ciclo de vida: Se detalla cuál fue el proceso seleccionado para la solución del problema, indicando en la misma, el ciclo de vida elegido, cómo el mismo fue llevado a cabo en el proyecto y cómo se fue desarrollando.

Ingeniería de requerimientos: Se aborda el proceso de ingeniería en requerimientos, la identificación de los requerimientos, su priorización, y el estudio sobre el alcance de los mismos

Arquitectura: Se detalla la arquitectura definida para la aplicación, así como sus especificaciones y las tecnologías utilizadas.

Proceso de desarrollo: En este capítulo se muestra cómo fue la organización del desarrollo del producto

Gestión del proyecto: Se menciona la estrategia de gestión de proyectos seleccionada, la planificación, el seguimiento de actividades, la gestión de riesgos y análisis de métricas tomadas.

Aseguramiento de Calidad: Se presentan los objetivos de la calidad, las actividades de aseguramiento de calidad que se realizaron, junto al plan de pruebas y detalles de métricas relativas a la calidad del proyecto.

Gestión de la Configuración: Se identifican los distintos elementos de configuración, se detalla el análisis realizado para elegir el repositorio y se define el proceso de control de configuración.

Conclusiones: Se enumeran las conclusiones que se extraen al finalizar el proyecto, junto a las lecciones aprendidas.

Bibliografía: Se detallan las referencias bibliográficas consultadas para realizar este documento.

Anexos: Se presentan los anexos generados para el presente documento

2. Planteamiento del problema

2.1. Contexto del problema

2.1.1. Problemática general

En la actualidad, C.U.C.T.S.A. dispone de aplicaciones de escritorio para el manejo de información referente a la administración de la empresa en su totalidad, utilizadas en las oficinas para la labor del día a día por parte de los empleados. Las mismas incluyen la gestión de liquidaciones, informes de novedades (faltas, suspensiones, horas extras, etc.), administración y asignación de horarios de viajes y servicios, gestión del personal (nuevos ingresos, cambios de cargo), consultas necesarias para el apoyo al servicio de atención al cliente, entre muchas más.

Pero las soluciones que ofrecen son muy limitadas a los trabajos de oficina, y mucha información que se maneja en la calle, terminales y ómnibus son realizadas y gestionadas de forma manual. Esto genera que las mismas no se digitalicen, aumente la dificultad para su transporte y visualización, y que sean duplicadas. En consecuencia, dicha información no se encuentra disponible de forma oportuna para otros empleados e incluso para clientes (como son los pasajeros de ómnibus, que no tienen la información actualizada sobre cancelaciones de viajes o cambios de horarios).

A su vez, existen oportunidades de mejora con respecto a la comunicación, tanto para el intercambio de comunicados, avisos y solicitudes, entre departamentos de la empresa, e incluso hacia los empleados.

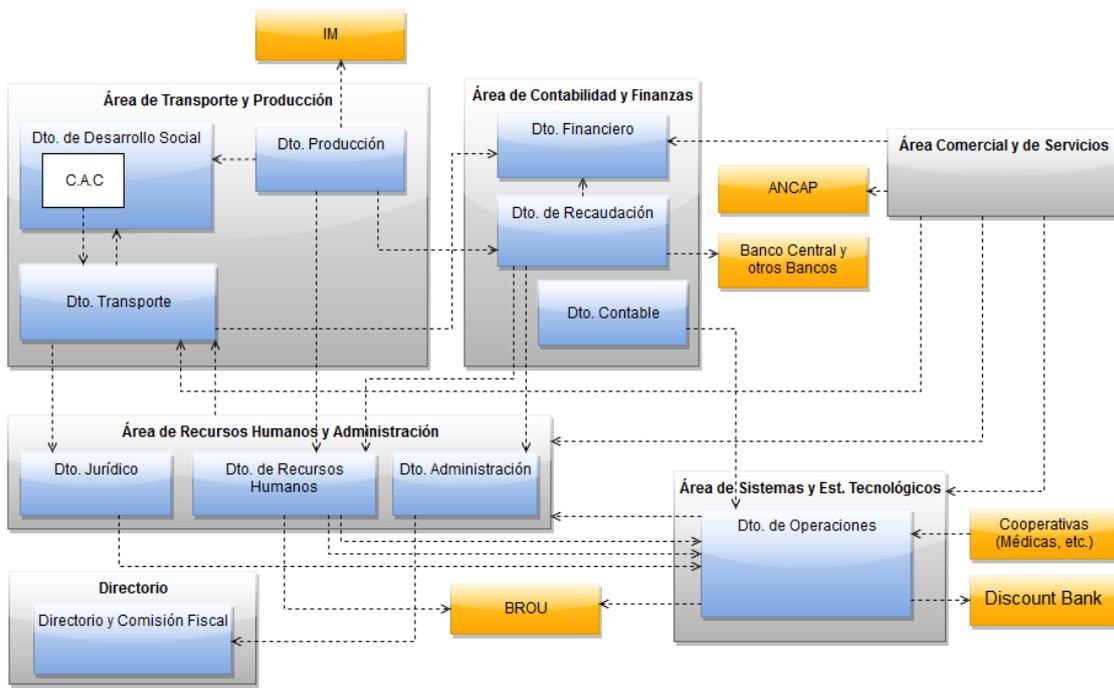


Ilustración 2-1 Intercambio de información entre las distintas áreas y departamentos de C.U.T.C.S.A.

La ilustración anterior es un diagrama simplificado de las comunicaciones internas de la empresa analizadas al inicio del proyecto. La misma muestra con flechas el envío de información desde un Área o Departamento de C.U.T.C.S.A. (de color gris y azul respectivamente), o empresas externas (de color amarillo), hacia el resto. Dicha información es enviada, según cada caso, a través de comunicados, correo electrónico, diskette, llamadas telefónicas, un sistema interno o por camioneta. Para visualizar más detalles sobre el intercambio de información entre departamentos y áreas de C.U.T.C.S.A., y otras empresas involucradas, puede ir al [Anexo 1 - Situación actual de la empresa](#).

2.1.2. Problemas abordados en el proyecto.

Esta sección tiene como objetivo describir de forma más detallada cómo son ejecutados algunos de los procesos de la empresa previos a la implementación del sistema que se describe en el presente documento. Los procesos a detallar son los realizados por empleados que desempeñan el rol de Inspector, Largador, Conductor, Guarda y Conductor Cobrador, respectivamente.

Existen dos tipos de controles que realiza el inspector: uno que se realiza en las paradas de ómnibus para el control del horario de los servicios que pasan por ella; y otro que se realiza sobre el ómnibus para el control del viaje.

A la hora de realizar el control del horario de los servicios que pasan por la parada, el inspector lleva consigo una planilla que describe los servicios, la cual completa si se fue respetando dicho horario.

En el control que se realiza sobre el ómnibus, el inspector debe esperar el mismo en la parada, y al subir imprime un reporte desde la máquina expendedora de boletos. Dicho reporte contiene la información esencial del viaje y servicio (identificadores del servicio y variante, horario de salida y llegada del viaje, boletos vendidos, boletos anulados, horarios de pasada por los distintos puntos de control, coche y funcionarios asignados, entre otros detalles). El inspector corrobora que dichos datos se estén cumpliendo y realiza un control de los pasajeros a bordo, pidiendo el boleto a cada uno y verificando que se encuentre en el reporte. Al finalizar la inspección, guarda dicho reporte (y todos los que imprimió en la jornada) para ser entregado en la oficina. Los reportes son luego analizados por los jefes del departamento.

Cuando un servicio o viaje es modificado por un Largador (ya sea porque se retrasó o rompió el ómnibus asignado, retrasos del personal o cualquier otro inconveniente que no permita que un viaje se realice a la hora planificada), el mismo no se registra en ningún sistema en tiempo y forma para ser considerado en tiempo real. Esto genera que la información se encuentre desactualizada, principalmente para clientes de C.U.T.C.S.A. (como son los pasajeros, quienes no llegan a tomar conocimiento de la suspensión o retraso de un viaje).

Los conductores, guardas y conductores cobradores deben concurrir todos los días a las oficinas de recaudación para que se les entregue la asignación del servicio para la siguiente jornada, o bien llamar por teléfono a la oficina para conocerlo.

A su vez, todo empleado debe ser notificado mediante llamada telefónica o concurrencia personal sobre citaciones o vencimientos (como del carné de salud o de la licencia de conducir).

Por lo tanto, existen oportunidades de mejora más particulares con respecto a comunicación de información, notificaciones y modificaciones que hacen referencia al personal que trabaja en la calle, que al día de hoy no se encuentran disponibles.

Si se desea ver con más detalle los problemas y oportunidades de mejora para la empresa, detectados al inicio del proyecto, puede ir al capítulo Problemática Actual del [Anexo 1 - Situación actual de la empresa](#).

2.2. Partes interesadas del proyecto

Se categorizaron los distintos interesados según la necesidad a la cual responden: estratégica, táctica u operativa.

Se pudo detectar que los altos mandos de la empresa tienen una necesidad de posicionar la organización estratégicamente. En particular, se mantuvo contacto con el **Sub Gerente General**, quien manifestó la necesidad estratégica de innovación en la organización para poder así posicionarse y estar a la vanguardia frente a los competidores. Dado que el proyecto desarrollado brinda innovación sobre todo desde el punto de vista tecnológico, se estaría ayudando a cubrir parte de esta necesidad convirtiéndolo en uno de nuestros interesados principales.

A nivel de los departamentos de la empresa, hay necesidades de nivel táctico. El sistema desarrollado apuntó a incrementar la eficiencia, eficacia y control de varias actividades, con lo que se estaría ayudando con dichas necesidades. Las funcionalidades desarrolladas alcanzaron a empleados que pertenecen en su totalidad al Departamento de producción. Convirtiendo al **Jefe de departamento de Producción** en un interesado.

Un escalón por debajo, se encuentran los empleados que se lograron alcanzar con el desarrollo, los cuales tienen diferentes funciones: Inspector, Largador, Conductor, Guarda y Conductor Cobrador. Estos tienen necesidades de índole operativo, y con el sistema desarrollado se estaría ayudando en varias de ellas, como ser, facilitando la realización de tareas y aumentando el acceso a la información. Es por esto que fueron considerados como interesados en el proyecto. Dado que también van a ser los usuarios directos de la aplicación, es oportuno describir con mayor detalle a cada uno de ellos:

Inspector, es quien se encarga del control de los servicios de los coches que se encuentran circulando, es decir, que se respete el horario asignado, control del estado del coche y del servicio, presentación del personal, y control de los pasajeros. A su vez, debe controlar la regulación de los servicios y realizar ajustes a los mismos por eventos e imprevistos (como accidentes de tránsito).

Largador, es aquel usuario que debe poner en funcionamiento los servicios a partir de la planificación, y asignar el personal que realizará los mismos para cada jornada laboral.

Conductor, cuya única función es manejar el coche, **Guarda**, quien solamente realiza la entrega de la venta de boletos a los pasajeros para la realización del viaje, y **Conductor Cobrador**, quien realiza las funciones del Conductor y Guarda al mismo tiempo. Dichos usuarios son los que realizan su labor sobre el coche, respetando los servicios asignados, es decir, concurrir al lugar de citación para la jornada en el lugar y hora definido, y realizar los viajes respetando los horarios y variantes.

Los usuarios del sistema descritos anteriormente hacen referencia a quienes utilizarán la aplicación realizada para el presente proyecto de fin de carrera. Como se mencionó previamente, este proyecto fue desarrollado con el objetivo de ser extendido para todo el personal de C.U.T.C.S.A., quedando como responsabilidad de la empresa continuar con dicho objetivo.

A lo largo del proyecto se mantuvo un contacto constante con el departamento de T.I. y principalmente con el **Gerente de Sistemas y Estudios Tecnológicos** de la empresa, quien fue considerado como el *Sponsor* del proyecto.

2.3. Principales aspectos de la solución propuesta

Luego de una gran cantidad de reuniones con actores claves pertenecientes a C.U.T.C.S.A. se logró no solo entender las distintas problemáticas de la empresa, sino que también se logró transformar las distintas necesidades detectadas en soluciones y funcionalidades concretas.

A continuación, se detalla en un esquema a alto nivel de la solución propuesta, en donde se puede visualizar también las principales interacciones que el mismo tiene con su entorno.

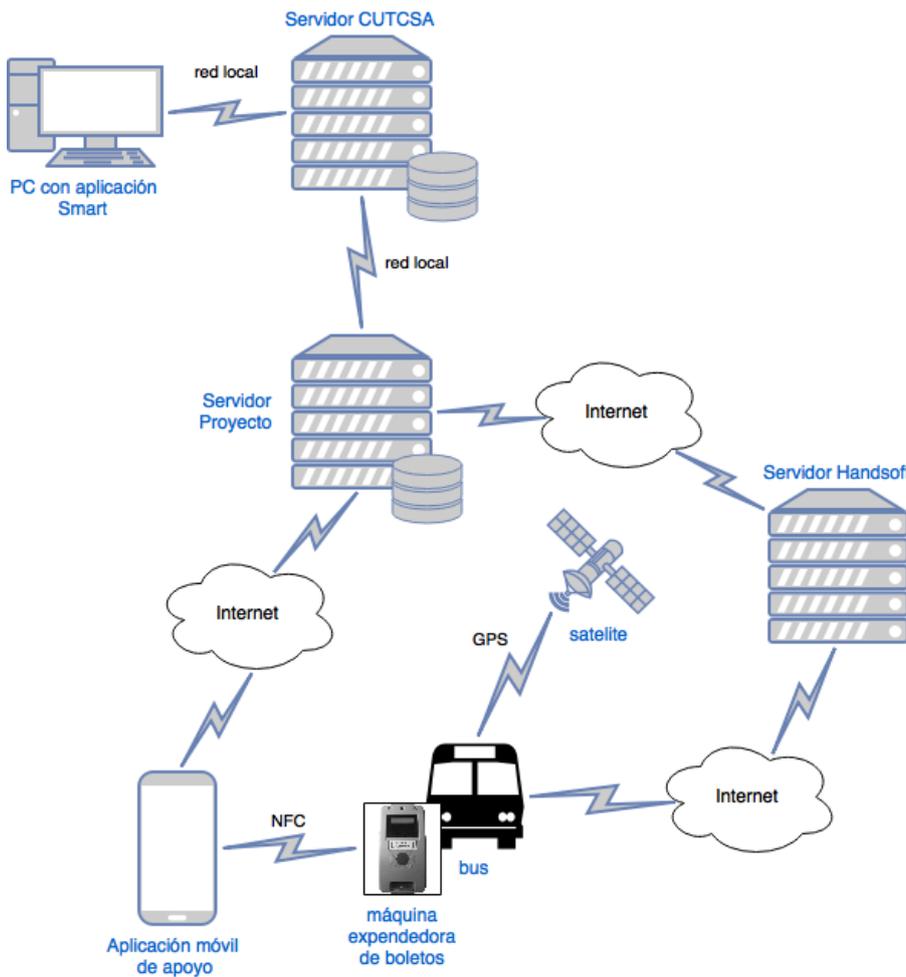


Ilustración 2-2 Diagrama general de la solución

Desde la empresa C.U.T.C.S.A. se ingresan datos e información a una base de datos interna mediante un sistema de escritorio llamado “Smart”. De estos datos se toman solamente los que van a ser utilizados por la aplicación móvil y son replicados y mantenidos en caso de cambios, en un servidor, en donde también se encuentran los servicios que van a dar soporte dicha aplicación.

Por otro lado, varias de las funcionalidades realizadas requieren la posición satelital de los coches. La misma se obtiene mediante un dispositivo GPS colocado en cada uno de ellos y es transmitida a un servidor, administrado por una empresa externa que es la encargada de mantener y exponer dichos datos mediante servicios. Esta información es consultada cuando se requiere, mediante el servidor del proyecto.

La aplicación móvil de apoyo opera recibiendo y enviando datos exclusivamente con el servidor dedicado para la misma.

Para poder realizar un control de servicio al momento de abordar un inspector un coche, la aplicación móvil obtiene los datos necesarios para el control utilizando una tarjeta que oficia de puente entre la expendedora de boletos del coche y el dispositivo móvil del inspector. Esta comunicación se realiza mediante tecnología NFC.

Se decidió que la solución propuesta se base en una aplicación móvil debido a varios factores. Partiendo de los problemas mencionados en el [capítulo 2.1 Contexto del Problema](#), existe una falta de digitalización de información, y su disponibilidad se encuentra solamente mediante papel o computadores de escritorio. En contraste, más del 70% de los empleados pasan gran parte de su jornada laboral fuera de las instalaciones de la empresa, en donde no se cuenta con acceso a sistemas informáticos. De esta forma, el desarrollo de una plataforma móvil fue una solución ideal.

La aplicación cuenta con varios grupos de funcionalidades comunes a distintos roles de usuarios, además de funcionalidades generales:

- Las generales, son funcionalidades básicas para gestionar la sesión de los usuarios, así como también la recepción y manejo de notificaciones y noticias.
- Las relacionadas a los inspectores son las referentes al control de servicios, consultas varias relacionadas a los coches y sus posiciones, horarios de servicios, información de paradas y puntos de interés.
- Las relacionadas a guardas, conductores y conductores cobradores son las funcionalidades referentes con la consulta de próximas asignaciones de servicios.
- Las relacionadas a los Largadores son las funcionalidades referentes a la gestión de servicios.

2.4. Supuestos y restricciones

A continuación, se presentan los supuestos y restricciones relacionados al producto y su contexto.

2.4.1. Supuestos

- La empresa va a brindar los dispositivos móviles necesarios para realizar las pruebas y desarrollo del sistema.
- Se tendrá acceso a la empresa, durante horario laboral, para que el equipo pueda reunirse con interesados al proyecto, además de para poder realizar pruebas y desarrollo del sistema.
- Se contará con acceso a los sistemas del cliente para poder realizar pruebas de validación.

2.4.2. Restricciones

- Hay zonas de la ciudad con mala recepción, en donde puede suceder que los coches pierdan señal del GPS y/o GSM.
- Duración del proyecto, ya que el proyecto tiene una duración máxima de un año.
- Dedicación del equipo. Debido a obligaciones personales, el equipo tiene una dedicación estimada de 15 a 20 horas semanales al proyecto.

3. Proceso y ciclo de vida

En el presente capítulo se explican las principales características de las metodologías utilizadas, sus conceptos teóricos y el marco de trabajo para llevar a cabo el proyecto.

3.1. Ciclo de vida

A la hora de elegir el ciclo de vida, se presentaron varias preguntas para saber cuál se adapta mejor a las necesidades del proyecto, y se evaluaron en base a las características del mismo. Por esta razón, fue necesario comprender cuáles eran las características del proyecto más relevantes

Las características más destacadas son:

Ciente de tamaño importante con principales interesados en cargos demandantes: Esto implica que las personas más interesadas en el proyecto no son fácilmente accesibles a la hora de validar, relevar requerimientos y evacuar dudas. Por dicha razón, la comunicación se realizaría por medio de correo electrónico y reuniones esporádicas, siendo de esta manera que la documentación juegue un rol importante en la comunicación para evitar así ambigüedades sobre lo comunicado.

Aplicación seguirá siendo desarrollada por personas ajenas al proyecto: Al terminar el proyecto, toda la información relevada y las decisiones sobre la aplicación y el código fuente serán entregadas al cliente. Por esta razón, todo debe estar debidamente documentado para que se pueda continuar en el desarrollo de la aplicación sin la participación de ninguno de los involucrados en el desarrollo de la misma.

Poco conocimiento del cliente sobre cuál es el futuro de la aplicación: El cliente desea tener una aplicación para un rol de empleado particular y algunas de las necesidades que la aplicación podría cubrir. Pero esta solo cuenta con pocas funcionalidades y solo cubre las necesidades de un rol. Para el resto de los empleados de la empresa el cliente no conoce de qué forma lo puede ayudar una aplicación. Para poder tener una mejor perspectiva del problema de la empresa, se realizó una investigación en todas las áreas relevantes para poder identificar las necesidades no cubiertas y cuáles podrían ser mejoradas a través de la aplicación.

Tiempo acotado: el proyecto debe realizarse en un año de trabajo y el plazo no podría extenderse, por lo que la duración del mismo está acotada

Disponibilidad de horarios variada: A pesar de ser un equipo de trabajo de tres personas, la disponibilidad de horarios es muy variada entre los integrantes. Por lo

tanto, es poco probable que exista la posibilidad de tener reuniones diarias y compartir un mismo espacio de trabajo.

Conocimiento de la tecnología: El proyecto está orientado a una aplicación Android con un servidor en tecnologías .NET, por lo que desde el comienzo del proyecto la mayoría de las tecnologías a utilizar son conocidas y cada integrante del equipo tiene experiencia en al menos una de ellas.

Dadas las características anteriores se decidió realizar un ciclo de vida Evolutivo (ver ilustración 4.1). La razón principal sobre la cual se elige el mismo está relacionada con la naturaleza de los requerimientos. Al no estar definidos desde el comienzo del proyecto, esta metodología permite incorporar nuevos requerimientos luego de cada evolución. Esto, a su vez, permite que exista una etapa formal de relevamiento que se podrá documentar adecuadamente para luego ser validada con el cliente. Asimismo, le es útil al cliente para poder reconocer un avance en la aplicación y así poder comprender y visualizar el producto final.

Obtener todos los requerimientos al comienzo del proyecto es difícil con cada una de estas características, no solo por la dificultad de que el cliente no tiene claro cuáles son sus ideas para el proyecto, sino también por la dificultad que tiene para transmitirlos correctamente, y por la evolución de los requerimientos a medida que avanza el proyecto. El modelo de ciclo de vida evolutivo afronta este problema mediante una iteración de ciclos -requerimientos-diseño-desarrollo-prueba [20].

El principal riesgo de este ciclo de vida es la posibilidad de una arquitectura cambiante, dado que se relevan nuevos requerimientos en cada evolución. Para minimizar este riesgo se optó por realizar una arquitectura modular, que se encuentra detallada en el [capítulo de Arquitectura](#).

Otro beneficio que se obtiene es la habilidad de poder estimar mejor cada evolución en base a los datos obtenidos de los anteriores y, teniendo en cuenta los requerimientos a desarrollar, que el equipo pueda adaptarse mejor al plazo final de la entrega.

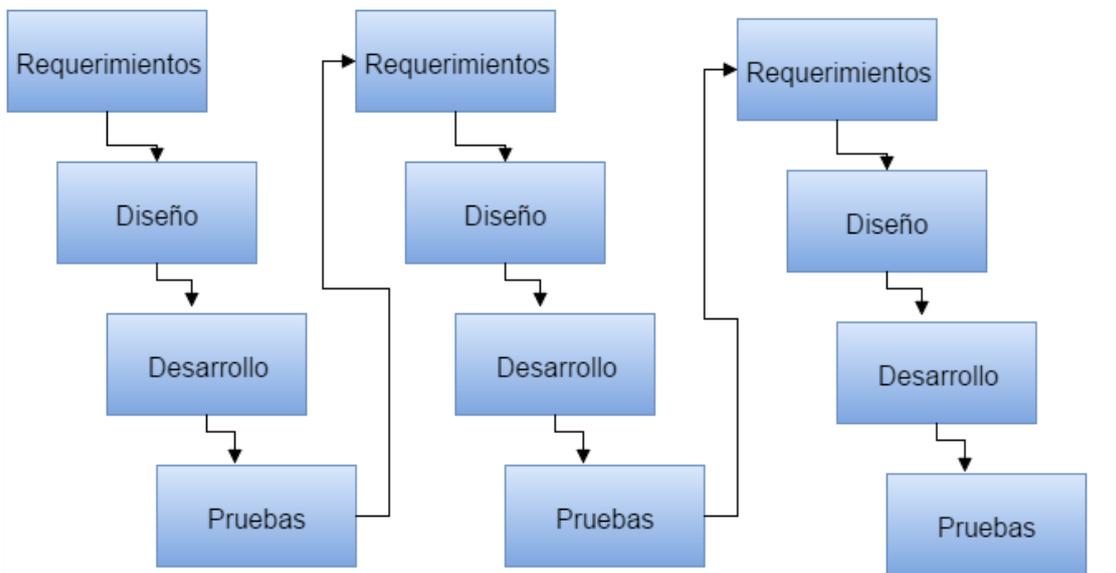


Ilustración 3-1 Ciclo de vida evolutivo

3.2. Metodologías de referencia

El proyecto se desarrolló con una metodología tradicional mediante una gestión basada en planes, en donde los procesos aplicados en el proyecto se realizaron en forma secuencial y cuentan con un activo como resultado. A continuación, se listan los procesos con sus correspondientes activos

Proceso	Activos
Ingeniería de requerimientos	Especificación de requerimientos
Diseño	Documento de Arquitectura
Desarrollo	Aplicación móvil, Aplicación del servidor con Base de Datos. Solución de seguridad para ambas plataformas
Pruebas	Plan de pruebas
Aseguramiento de Calidad	Plan de calidad y Plan de SQA
Gestión de la configuración	Plan de SCM
Gestión de proyecto	Plan de proyecto, Informes de avance entregados al cliente.

Tabla 3-1 Procesos y sus activos

4. Ingeniería de requerimientos

4.1. Introducción

En el presente capítulo se detalla el proceso de Ingeniería de Requerimientos utilizado, describiendo en profundidad cómo se realizó cada una de sus etapas: Relevamiento, Análisis, Especificación y Validación. Se explican también los criterios utilizados para priorizar los requerimientos relevados y la estructura utilizada para especificarlos.

4.2. Proceso de Ingeniería de Requerimientos

Las etapas del proceso de Ingeniería de Requerimientos pueden verse en la siguiente ilustración.

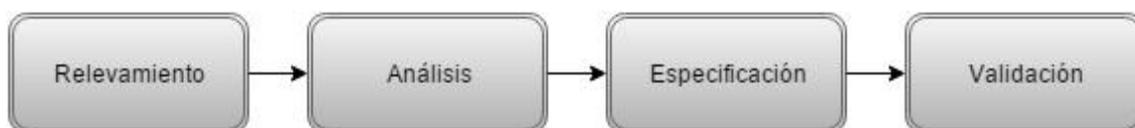


Ilustración 4-1 Etapas del proceso de ingeniería en requerimientos

Al comienzo de la primera evolución del proyecto, este ciclo se ejecutó de forma completa, es decir que se ejecutaron las cuatro etapas del proceso. En las siguientes evoluciones solo se hizo de forma parcial, partiendo de la etapa de la Especificación. Esto fue debido a que, con el relevamiento realizado inicialmente, se logró abarcar todos los sectores de la empresa que se contemplaron con la aplicación desarrollada.

El proyecto a realizar contaba con la complejidad extra de que 2 de los 3 integrantes del equipo no tenían conocimientos del negocio. Y el que sí lo tenía tampoco conocía en profundidad todas las áreas del mismo. Por lo que se estipuló que era muy importante, además de relevar requerimientos, poder entender cómo funciona la empresa, cómo es su estructura organizacional, cómo son sus procesos y los diferentes roles que hay en la misma. De esta manera, luego de entender cómo funciona, poder hacer un relevamiento más eficiente.

Previo al relevamiento, el equipo reunió información de la empresa de distintas fuentes, incluyendo un organigrama e información de algunos procesos.

Luego de obtenida la información se analizó la misma y se realizaron reuniones internas del equipo, en donde principalmente el integrante que trabaja dentro de C.U.T.C.S.A. ayudó a los restantes a evacuar dudas y comprender la empresa, su composición,

terminología interna y demás, pudiendo así nivelar el conocimiento que se tenía de la empresa por parte de los integrantes.

4.2.1. Relevamiento

Previo a realizar el relevamiento y contando los 3 miembros del equipo ya con un mayor conocimiento sobre la empresa, se definieron una serie de reuniones con actores claves dentro de la misma. Como primera decisión, se estipuló en una primera etapa elegir empleados con un mando medio, es decir, gerentes o jefes de departamentos. Esto fue debido a que se entendió que estas personas podrían tener una visión global de sus departamentos, pero enfocada a sus respectivas áreas.

Se pueden ver en la siguiente Tabla los distintos empleados entrevistados durante la etapa de relevamiento y su cargo correspondiente dentro de la empresa.

Nombre	Cargo
Fernando Romero	Jefe de Dpto. de Producción
Carlos Miglino	Gerente. de RRHH y Administración
Amador Lozano	Jefe de Dpto. de Desarrollo
Miguel Pose	Jefe de Dpto. de Operaciones
Néstor Silva	Jefe de Dpto. de Transporte
Washington Filippini	Jefe de Dpto. de Recaudación
Raquel Pioli	Gerente Comercial y de Servicios

Tabla 4-1 Empleados de la empresa y su correspondiente cargo

En la siguiente ilustración se puede ver un organigrama simplificado de la empresa, circuladas en rojo los sectores de la empresa donde se realizaron entrevistas.

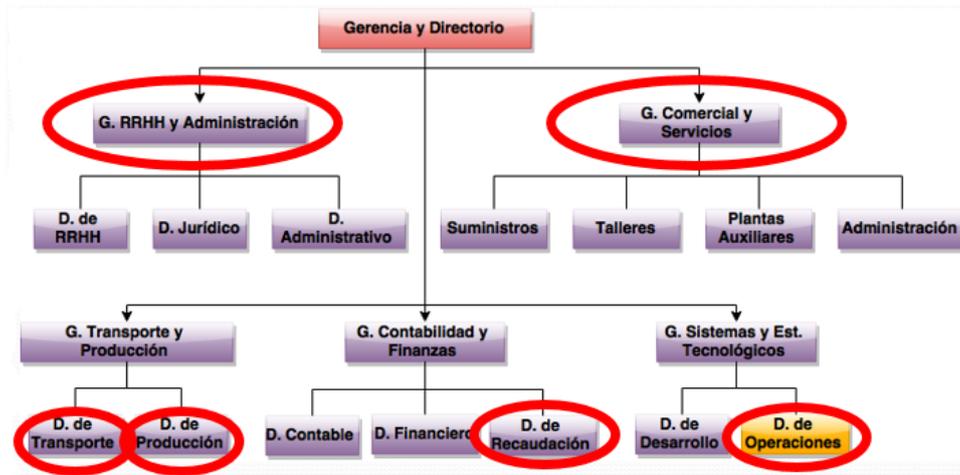


Ilustración 4-2 Organigrama de la empresa

Con dichas entrevistas se buscó continuar aprendiendo sobre la empresa, entendiendo los procesos y terminología interna. Y, por otro, lado descubrir necesidades y posibilidades de mejoras en procesos, para así poder convertirlos en requerimientos para el proyecto.

Los relevamientos fueron realizados principalmente mediante reuniones presenciales, a las cuales siempre se buscó que asistiera el integrante que trabaja en la empresa y uno de los 2 integrantes del equipo que eran externos a la misma. De esa manera, se pudo tener una visión diferente a la del integrante con mayor conocimiento del negocio, y para que, al momento de realizar el análisis, la información recabada sea lo suficientemente clara para poder ser entendida y analizada por todos los integrantes del equipo.

Otra forma de relevar requerimientos fue solicitar diagramas de flujo, definición de procesos y manuales del negocio. En menor medida, también se realizaron consultas puntuales mediante correo electrónico.

Es oportuno aclarar, que antes haber tomado contacto con la empresa, ya existía en la misma la motivación de realizar una aplicación móvil que atendiera diversas necesidades de los inspectores. Por lo que algunas necesidades para dichos usuarios ya estaban identificadas, aunque no con una documentación formal. Por dicha razón, en el relevamiento realizado fueron tenidos en cuenta, y se enfocó en entender todos los procesos relacionados a los mismos, ya que se sabía que se iban a implementar. Estos requerimientos puntualmente fueron los relacionados al control de un servicio y mostrar arribo de líneas a una parada.

Finalmente, para poder relevar requerimientos no funcionales (RNF), se realizó una reunión con el Desarrollador de mayor experiencia en la empresa, el Administrador de base de datos (DBA) y con el Administrador de redes.

4.2.2. Análisis

En la etapa del análisis se realizaron 3 actividades.

Se comenzó transcribiendo cada una de las reuniones en documentos individuales y luego se realizaron diagramas de flujo de los procesos principales identificados.

Se enviaron dichos documentos y diagramas a cada uno de los entrevistados para su validación, corrigiendo los documentos y diagramas cuando hubo acotaciones o diferencias con los entrevistados. Se decidió hacer esta validación previa a la validación final de los requerimientos, dado que el equipo tuvo el doble trabajo de entender el negocio y recabar requerimientos al mismo tiempo, evitando así retrabajo en caso de no haber entendido algo correctamente en futuras etapas.

Finalmente, se analizaron todos los documentos y diagramas mencionados anteriormente, y se resumió en un documento único, que se encuentra en el [Anexo 1 - Situación actual de la empresa](#), el cual contiene las características principales de la empresa, detalle de qué hace cada área de la misma y descripción de algunos de los procesos, diversas problemáticas identificadas en cada uno de ellos, oportunidades de mejora y finalmente posibles requerimientos que atienden a esas oportunidades, descriptos a alto nivel.

Luego de confeccionado el documento, se organizó una reunión con el Sub Gerente General de la empresa poder mostrarle todo lo recabado hasta el momento y que, con su visión global de la empresa, pudiese validarlo previo a la etapa de especificación formal.

4.2.3. Especificación

Si bien al comenzar esta etapa ya se tenía una definición a alto nivel de los posibles requerimientos del sistema, en base a esas definiciones se confeccionó un documento de Especificación de Requerimientos (ESRE), el cual contiene una descripción detallada de los requerimientos a implementar, definición de casos de uso asociados, con prototipos de interfaz de alto nivel, realizados a mano alzada para evitar que el cliente confunda estos prototipos con el producto final.

Este ESRE es evolutivo, y fue creciendo al comienzo de cada evolución, agregándose los requerimientos nuevos a implementar.

El ESRE final completo se puede ver en el [Anexo 2 - ESRE](#).

4.2.4. Validación

Además de las validaciones intermedias realizadas con cada uno de los entrevistados, y luego con el Sub Gerente General de la empresa, al finalizar la especificación de requerimientos al comienzo de cada evolución, se validó la misma con el Gerente de Sistemas y Estudios Tecnológicos, con quien se tuvieron reuniones de forma constante.

Dicho gerente, además, se encargó de validar la Especificación de Requerimientos para realizar cambios en los mismos. Se utilizó un versionado para cada Caso de Uso, para así evitar tener que leer todo el documento en cada revisión si no era necesario.

Los requerimientos que se especificaron para las evoluciones siguientes a la primera, terminaron estando relacionados con el Departamento de Producción, el cual se encarga de los inspectores, largadores, conductores, guardas y conductores cobradores. Por ello, se decidió que el Jefe de este Departamento validara el documento en cada una de estas evoluciones.

4.3. Estructura utilizada para la especificación

Para especificar cada caso de uso se utilizó la estructura que se muestra en la siguiente tabla.

RF asociado	C3-RF1
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF12, RNF13.
Actor	Guarda, Conductor o Conductor-Cobrador
Permisos para acceso a datos	Público
Precondición	<ul style="list-style-type: none">• Guarda, Conductor o Conductor Cobrador que haya iniciado sesión en la aplicación.• El dispositivo debe tener acceso a internet.• Tienen que estar cargadas las asignaciones de servicios en el sistema.

Tabla 4-2 Estructura de especificación de requerimientos

Seguido al encabezado del caso de uso mostrado en la tabla, se describe el curso de la funcionalidad paso a paso, agregando al mismo las interfaces de ejemplo de baja fidelidad realizadas a mano, como la que se muestra en la siguiente ilustración



Ilustración 4-3 Ejemplo de interfaz en caso de uso

4.4. Criterios de la priorización de los requerimientos

Para poder priorizar los requerimientos relevados, en primer lugar, se realizó una lista de todos los requerimientos asignándoles a cada uno una prioridad, respetando la siguiente escala:

- Muy baja
- Baja
- Media
- Alta
- Muy Alta

La prioridad fue asignada principalmente pensando en la cantidad de usuarios a los que afectaría y también en qué medida ayudaría a la empresa en general.

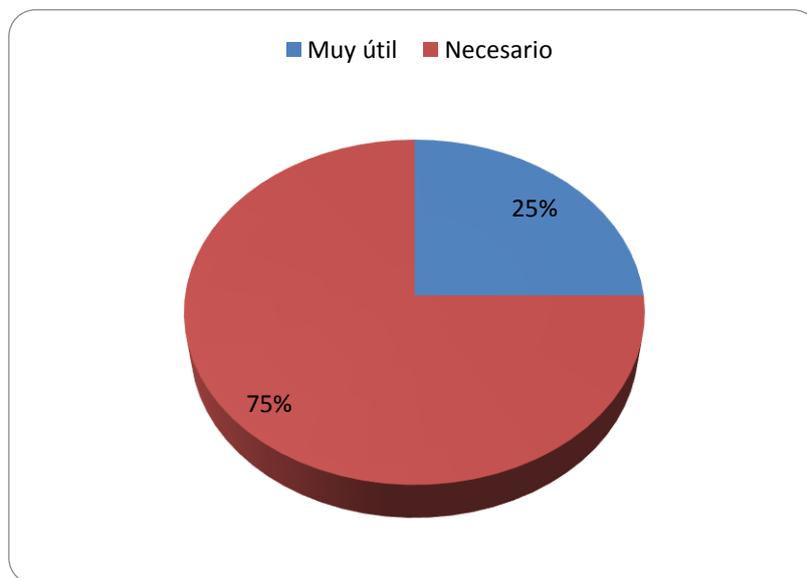
Al realizar la priorización se tuvo en cuenta que algunos de los requerimientos de los inspectores fueron clasificados como de Muy alta prioridad por el cliente.

Por otro lado, se involucró al Gerente de Sistemas y Estudios Tecnológicos quien, en paralelo a nuestra priorización, indicó qué requerimientos tenían más prioridad según su punto de vista.

En conjunto con el Jefe de Operaciones, se realizó una encuesta a un grupo representativo de futuros usuarios de rol Inspector, a los cuales, entre otras cosas, se les consultó sobre la utilidad de los requerimientos especificados referentes a sus tareas en la empresa.

Las opciones posibles de respuesta fueron: Nada útil, Poco útil, Muy útil, Necesario.

Se presentan a continuación en la siguiente gráfica, el resultado de uno de los requerimientos principales del sistema: Control de Servicio. En donde se puede observar que aproximadamente 75% de los encuestados lo considera Necesario y un 25% Muy útil.



Gráfica 4-1 Resultado de funcionalidad Control de servicio

Asignándole puntos del 0 al 4 a la escala definida anteriormente, y sumando las respuestas de cada entrevista, se logró tener una priorización por parte de los futuros usuarios del sistema. Se muestra a continuación una gráfica con los requerimientos priorizados según los puntos totales obtenidos.

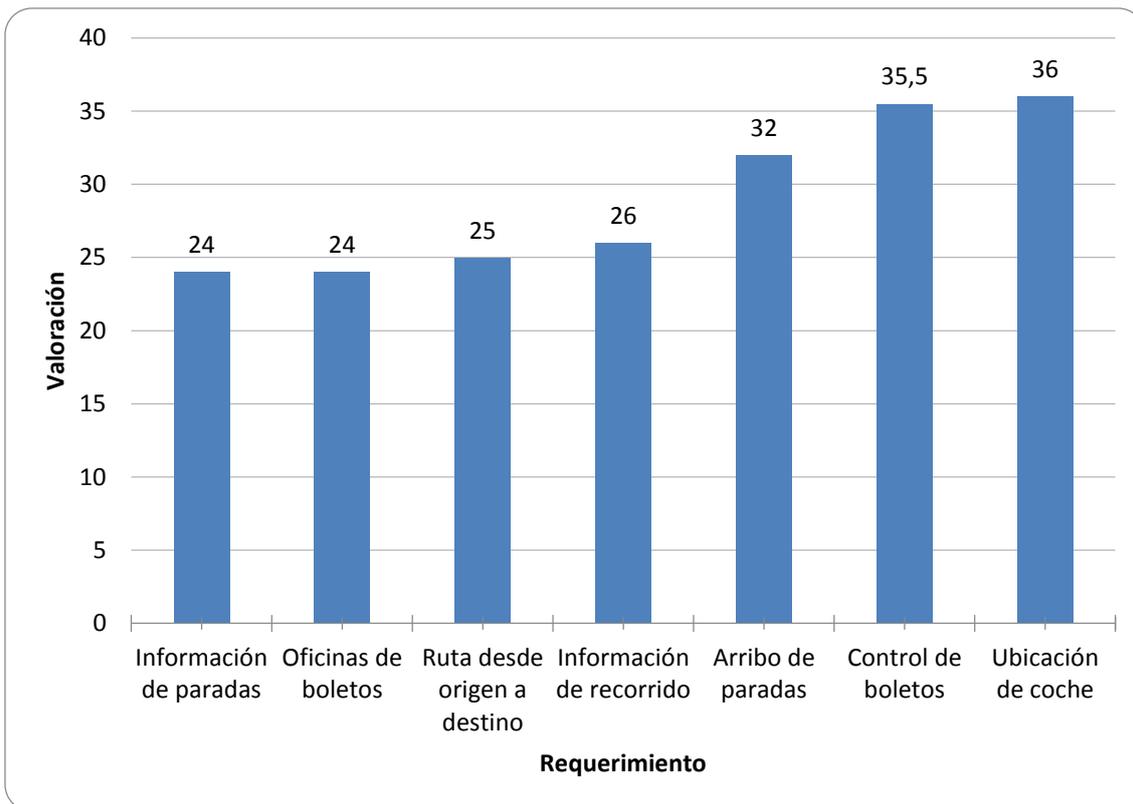


Gráfico 4-2 Requerimientos priorizados por encuestados

Para ver la encuesta completa realizada, ir al [Anexo 3 - Encuesta Inspectores](#).

Según la priorización realizada, las restricciones de disponibilidad del equipo y teniendo en cuenta las estimaciones realizadas utilizando la técnica de Puntos de Función (se puede ver más detalle de esto en [Capítulo Gestión de Proyecto](#)), se logró definir qué requerimientos desarrollar al comienzo de cada evolución. En la siguiente sección se presentan la totalidad de los mismos.

4.5. Requerimientos funcionales

Se listan a continuación todos los requerimientos funcionales del sistema, agrupados por los diferentes módulos.

Requerimientos comunes a todo el sistema, módulo “General”:

G-RF1: Log In: Se deberá iniciar sesión para poder utilizar cualquiera de las funcionalidades de la aplicación. Según el rol del usuario son las funcionalidades que se habilitarán para usar.

G-RF2: Log Out: Al terminar de utilizar la aplicación, los usuarios deberán de cerrar su sesión.

G-RF3: Mostrar Menú: Se deberá de poder acceder a un menú donde se tendrá acceso a todas las funcionalidades disponibles para el rol de usuario que inició sesión en la aplicación.

G-RF4: Reportar un error en la aplicación: Se deberá de poder reportar errores en la aplicación desde cualquier sección de la misma que se encuentre navegando.

G-RF5: Recepción de notificaciones: Recepción de notificaciones varias dentro de la aplicación, enviadas desde distintos sectores de la empresa y con destino a usuarios y/o roles específicos. Algunos ejemplos: aviso de asignación de servicio, vencimientos de carné de salud, vencimiento de libreta de conducir y nuevos desvíos planificados.

G-RF6: Gestión de notificaciones: Se deberá poder visualizar todas las notificaciones recibidas, de manera diferenciada si ya fueron leídas previamente o no. Poder acceder a las mismas, y eliminar las que no se deseen conservar en la aplicación.

G-RF7: Visualización de noticias: Recepción de noticias referentes a la empresa, las cuales se desplegarán por defecto al iniciar la aplicación, pudiéndose acceder a las mismas luego, mediante una opción dedicada en el menú de usuario.

Requerimientos Módulo “Inspectores”:

C2-RF1: Ver información de paradas: Ver las paradas más cercanas a la ubicación del usuario y mostrar información sobre las mismas (número, dirección y líneas que pasan por ella).

C2-RF2: Visualizar el recorrido de cierta línea en un mapa: Mostrar en un mapa el recorrido que tiene cierta línea, indicando en el mapa las paradas por las que pasa.

C2-RF3: Mostrar arribo de líneas a la parada: Mostrar los próximos arribos de líneas a una parada (pudiendo filtrar las búsquedas por parada). Para cada servicio próximo a arribar, se podrá ver los siguientes datos: tiempo estimado de llegada a la parada (utilizando GPS a bordo del coche) y ubicación en el mapa.

C2-RF4: Ubicar coche en un mapa: Ver la ubicación e información (número, recorrido, destino) de cierto coche en un mapa (utilizando la información provista por el GPS a bordo del coche).

C2-RF5: Ver ruta entre un origen y un destino: Conocer la ruta y qué ómnibus (solo de la empresa C.U.T.C.S.A.) tomar desde un origen a cierto destino. Se indicará a dónde se deberá caminar y a qué parada dirigirse-

C2-RF6: Ver información de puntos de interés: Ver información (nombre, dirección, teléfono y horarios) de las oficinas de venta de boletos, oficinas de línea y demás dependencias de C.U.T.C.S.A.

C2-RF7: Control del servicio: Obtener el reporte, que en la actualidad se obtiene impreso a través de la expendedora de boletos de un coche, conteniendo la información necesaria para la inspección del servicio. A su vez, poder realizar el control de boletos acercando la tarjeta del pasajero al celular y que el mismo alerte en caso de que la tarjeta no se haya utilizado en el viaje. En caso de que el pasajero no haya adquirido su boleto utilizando tarjeta, se podrá controlar mediante la numeración del boleto. Finalizado el control, quedara guardado un registro del mismo.

C2-RF8: Mostrar horario planificado de una línea: Ver el horario planificado de cada recorrido para cierta línea.

Requerimientos Módulo “Guardas/Conductores/Conductores-Cobradores”:

C3-RF1: Consultar servicios asignados: Consultar todos los servicios asignados al usuario para las próximas jornadas laborales.

Requerimientos Módulo “Largadores”:

C4-RF1: Modificar servicios: Poder modificar el horario de salida respecto al planificado y el coche asignado a un viaje.

C4-RF2: Alta servicio: Poder dar de alta un viaje para una fecha y horario determinada.

C4-RF3: Baja servicio: Poder dar de baja un viaje para una fecha y horario determinado.

4.6. Requerimientos no funcionales

4.6.1. Características de calidad

Se describen a continuación las características de calidad requeridas para el sistema, dados los distintos tipos de usuario y los objetivos generales del producto.

Para su definición el equipo se basó en la norma vigente a la fecha referente a requerimientos de calidad: ISO/IEC 25010:2011 [21].

RNF01: Fiabilidad – Disponibilidad: El sistema deberá de tener una disponibilidad mínima del 98%.

RNF02: Usabilidad – Facilidad de Operación: Debido a que esta aplicación va a ser una herramienta de trabajo diaria para algunos usuarios, es muy importante que la misma no entorpezca la realización del mismo. El sistema deberá de poder ser controlado con facilidad, por lo que se determinó que todas las operaciones puedan ser accedidas mediante 3 pasos (*clicks*) como máximo.

RNF03: Usabilidad – Estética de la interfaz de usuario: La interfaz de usuario de la aplicación deberá de ser atractiva e intuitiva al momento de realizar cualquiera de las operaciones disponibles. Es por dicha razón que se deberán respetar las pautas de diseño sugeridas por Android. [22]

RNF04: Usabilidad – Facilidad de Aprendizaje: Se plantea un tiempo máximo de 30 minutos para que los usuarios puedan comprender y ser capaces de utilizar las funcionalidades principales sin mayores inconvenientes.

RNF05: Eficiencia – Comportamiento temporal: En condiciones normales de uso del sistema, el mismo no debería de tener una demora mayor a 5 segundos para realizar la operación solicitada por el usuario.

RNF06: Eficiencia – Capacidad: La aplicación deberá de ser capaz de soportar a 5000 usuarios. Se llegó a este número sumando el total de empleados de la empresa. Si bien no se van a alcanzar a todos estos durante el proyecto, los planes de la empresa son extenderlo para todos sus empleados.

RNF07: Mantenibilidad - Modificabilidad: El cliente va a continuar con el desarrollo y extensión de la aplicación luego de culminado el presente proyecto, por lo que es de vital importancia que el sistema pueda ser modificado fácilmente y eficientemente. Es requisito contar con una especificación de casos de uso, diagramas de paquetes, despliegue, clases y componentes.

RNF08: Seguridad - Confidencialidad: Las distintas operaciones ofrecidas por la aplicación solo deberán de ser accedidas por usuarios cuyo rol corresponde con un rol habilitado para acceder a dicha operación.

RNF09: Seguridad – Autenticidad: Se deberá garantizar la identidad de quienes van a realizar cualquier operación ofrecida por el sistema.

RNF10: Seguridad – Integridad: Se deberá de garantizar la prevención del acceso y/o modificación de datos no autorizado a los datos de la aplicación y a la aplicación en sí misma.

4.6.2. Restricciones

RNF11: Desarrollo en Android: El sistema debe ser desarrollado para Android versión 4.0.3 y posteriores.

RNF12: Compatibilidad con Samsung Galaxy ACE 3: El móvil en el que la aplicación va a ser utilizado es de marca: Samsung, modelo: Galaxy ACE 3. Por lo que tendrá que ser totalmente compatible con el mismo, ya sea en términos de funcionamiento como de visualización.

RNF13: Desarrollo en .Net con C# para el *backend*: Es necesario que el *backend* de la aplicación esté desarrollado en tecnología .NET y más precisamente utilizando lenguaje C#.

4.7. Descripción de funcionalidad principal

A continuación, se detalla la ejecución de una de las funcionalidades principales del sistema, a modo de ilustrar el funcionamiento de la aplicación. Las impresiones de pantalla fueron obtenidas del producto final.

La funcionalidad seleccionada es: **C2-RF7: Control del servicio**, la cual solamente puede ser ejecutada por un usuario de rol “Inspector”.

En primer lugar, al abordar un inspector a un coche para realizar un control, utilizando una tarjeta *Mifare Classic* puente, se obtienen los datos necesarios desde la expendedora de boletos, cargándose los mismos mediante NFC a la tarjeta. Luego de haber iniciado su sesión en la aplicación, el Inspector, desde el menú lateral, selecciona la opción de “Control de servicio”.

Acerca la tarjeta puente a la parte trasera del móvil, y este obtiene los datos necesarios para el control del servicio, nuevamente a través de NFC, y son desplegados en pantalla. Para comenzar el control de tarjeta de los pasajeros a bordo, debe seleccionar en la opción de “CONTROLAR” (segunda y tercera impresión de pantalla de la siguiente ilustración). En caso de que haya habido cualquier tipo de error al leer los datos, se advierte al usuario como muestra la primera impresión de pantalla de la siguiente ilustración.

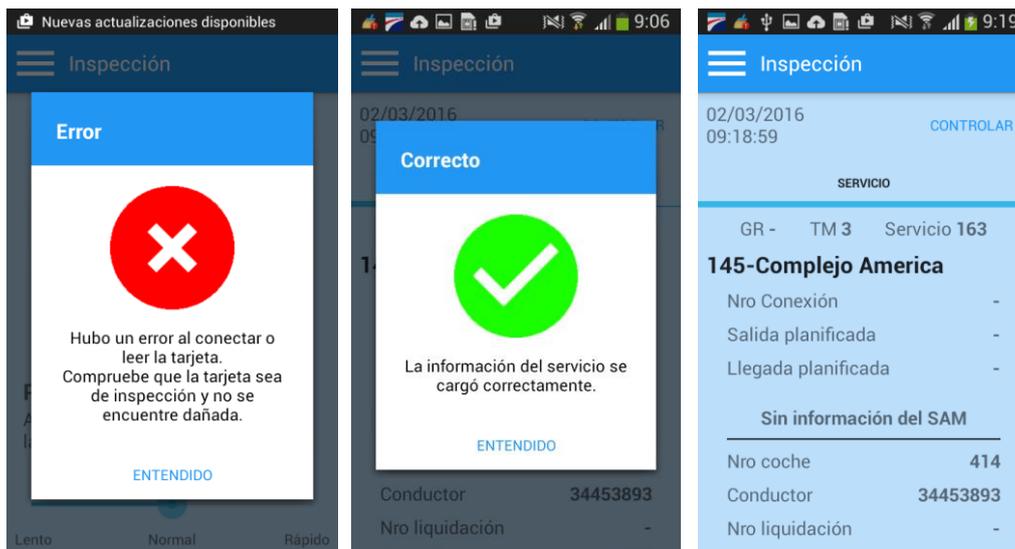


Ilustración 4-4 Datos generales del servicio.

Luego, procede a controlar a los pasajeros. Los que poseen boletos físicos de papel, se controla que la numeración pertenezca al rango de boletos vendidos desplegados en pantalla. Los que poseen tarjetas STM, las mismas son leídas mediante NFC acercándolas a la parte trasera del móvil. En la ilustración 4.5 se puede ver la numeración de boletos válida para el viaje y la lista de tarjetas que pasaron por la expendedora de boletos, donde se muestra con un color más oscuro la última tarjeta controlada.

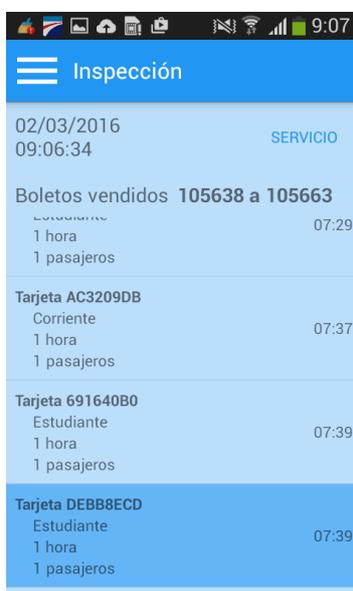


Ilustración 4-5 Detalle del servicio, pantalla de control.

La aplicación alertará al inspector si se trata de un viaje válido emitiendo un sonido corto de un *bip* y destacando la información de la tarjeta sobre la pantalla. En caso de que no sea válido (la tarjeta no estaba registrada en la expendedora de boletos) emitirá un

sonido de múltiples *bips* y se mostrará una alerta en pantalla. Para el caso de que la tarjeta ya se haya controlado previamente en el mismo control, la aplicación emitirá un sonido de múltiples bips y notificará al inspector del problema. En la ilustración 4.6 se pueden ver estos casos.

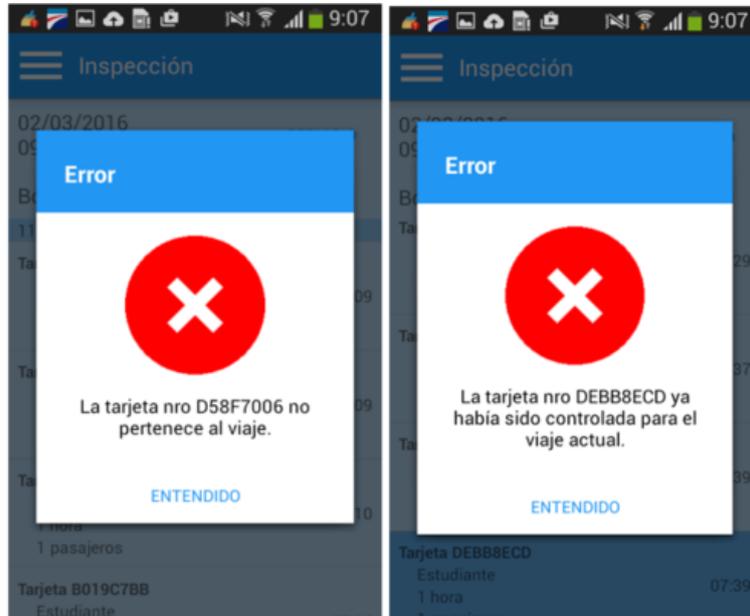


Ilustración 4-6 Casos de control de tarjeta errónea

4.8. Conclusiones

Se dedicó un esfuerzo de 46 horas en reuniones con el cliente durante la etapa de relevamiento, entendimiento del negocio y validación de la especificación de requerimientos.

Se puede concluir que el esfuerzo invertido en el proceso de ingeniería de requerimientos sirvió en un comienzo, para interiorizarse en el negocio y relevar requerimientos de forma más eficiente. Esto se tradujo a lo largo del proyecto en requerimientos estables durante las evoluciones realizadas y la minimización del tiempo de re trabajo debido a errores en los requerimientos.

Si bien al comienzo del relevamiento se entrevistó a personas pertenecientes a sectores de la empresa, de los cuales se relevaron requerimientos que no se llegaron a desarrollar, creemos que este tiempo no fue en vano, ya que nos ayudó a tener una visión global de futuros requerimientos posibles a agregar a la aplicación. Dado que el cliente va a continuar con la extensión de la misma, estos requerimientos no implementados, pero relevados, le van a facilitar dicha tarea.

5. Arquitectura

En el siguiente capítulo se describe la arquitectura del producto. Se realiza una especificación utilizando vistas arquitectónicas que representan diversos aspectos del sistema. En el mismo, también se detalla cómo se cumple con los requerimientos no funcionales y se fundamentan las decisiones sobre las tecnologías seleccionadas.

Contiene información acerca de cómo estos elementos se relacionan entre sí. De este modo, se crea una representación abstracta del sistema suprimiendo los detalles de los elementos que no afectan a la forma en que se utilizan, relacionan o interactúan entre sí.

5.1. Descripción General

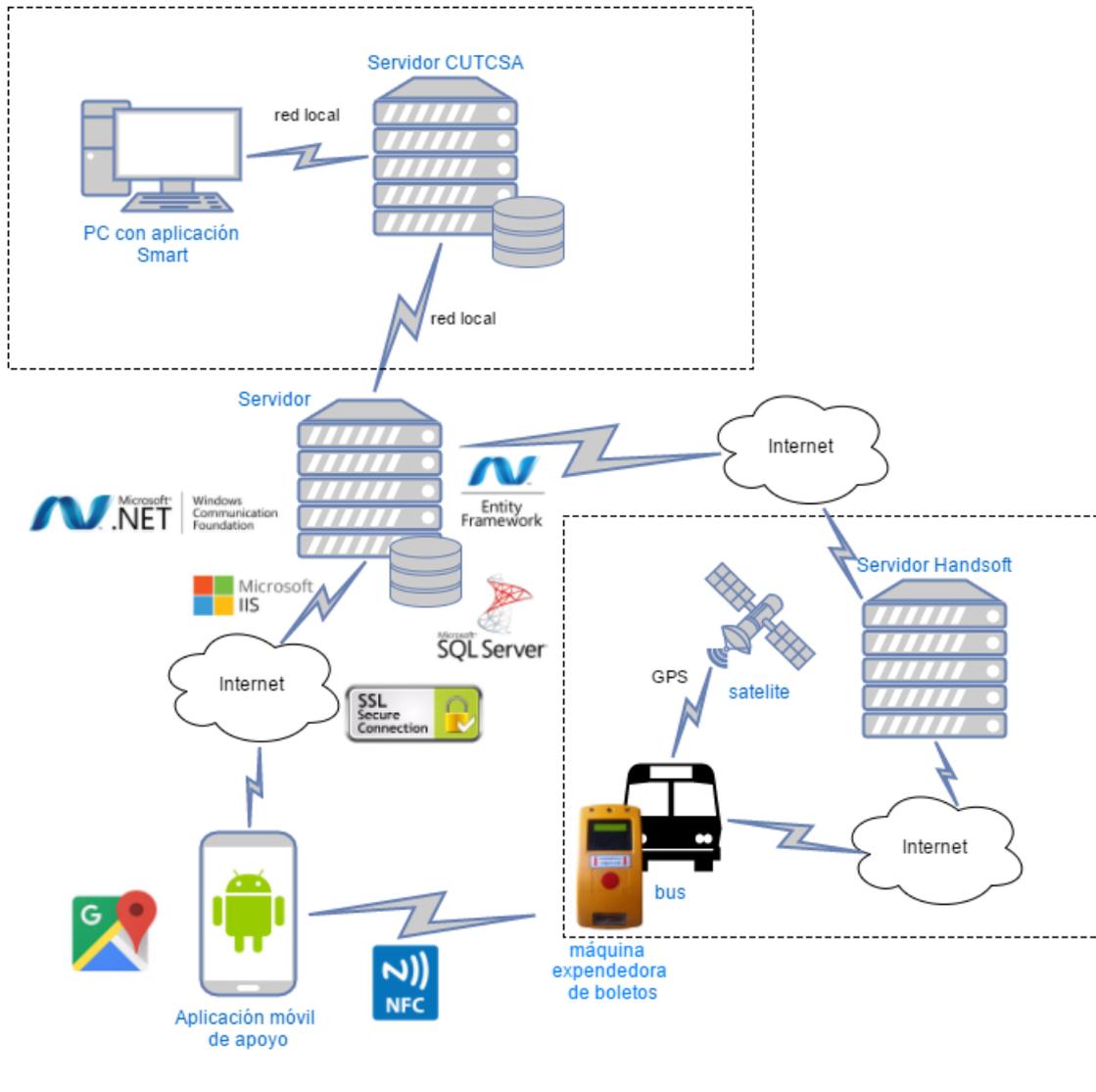


Ilustración 5-1 Diagrama de la Solución

En el diagrama anterior se pueden apreciar todos los elementos que construyen la solución al problema resuelto y qué tecnologías se aplican a los mismos. Las áreas encerradas en los rectángulos no forman parte de la solución desarrollada por el equipo del proyecto. Solo aquellas fuera de las mismas son las que fueron desarrolladas.

El principal actor entre los elementos es la aplicación móvil que se desarrolló sobre el sistema operativo Android. La aplicación es el único elemento de entre los indicados por el cual los usuarios tendrán acceso. Una de sus funcionalidades requiere la conexión con la máquina expendedora de boletos de la unidad de transporte, cuya comunicación se realiza a través de NFC, por medio de una tarjeta puente del tipo Mifare Classic. Otra de

las características importantes de la aplicación es el uso de *Google Maps* que forma parte de la solución de muchos de los requerimientos detectados.

Un móvil con sistema operativo Android va a ser provisto por la empresa a cada uno de los **inspectores**, pero no así para otros roles de usuarios que se pudieran incorporar en un futuro con el proyecto, teniendo estos que utilizar sus móviles personales.

El equipo investigó el porcentaje de utilización de los distintos sistemas operativos utilizados en *smartphones*, llegando a la conclusión de que un 82,8% de los usuarios utilizan Android y solamente un 13,9% utilizan iOS [23]. De esta forma, se decidió desarrollar también en la plataforma Android la aplicación correspondiente a los roles restantes que el sistema contempla: **largadores, cobradores, conductores cobradores y guardas**, para así poder abarcar a la mayor cantidad de usuarios posibles.

De este modo, se concluyó que la solución más óptima sería el desarrollo de una única aplicación móvil Android de apoyo a las tareas de los distintos funcionarios dentro de la empresa.

La aplicación no puede funcionar por sí sola por lo que establece una conexión con el servidor para obtener los datos necesarios para su funcionamiento. El servidor expone sus servicios mediante WCF (*Windows Communication Foundation*) y REST, y la comunicación se realiza utilizando SSL para mayor seguridad. Los servicios se publican a través de IIS

El servidor en sí funciona sobre una plataforma .NET, la cual maneja los datos comunicándose a través de *Entity Framework* con la base de datos en *SQL Server 2012*.

De los elementos que están fuera del alcance del equipo del proyecto, se encuentra el servidor de la empresa Handsoft, que brinda información sobre la posición del ómnibus y el tiempo de llegada de una unidad de transporte de un punto geográfico a otro. El primer servicio es requerido dado que C.U.T.C.S.A. no cuenta con la posición del coche, y solamente Handsoft tiene acceso y mantiene a esa información.

Información más detallada sobre la arquitectura se encuentra en el [Anexo 4 – Documentación de Arquitectura](#).

5.2. Arquitectura vs Atributos de calidad

A continuación, se describen los principales requerimientos no funcionales y los atributos de calidad más relevantes tomados en cuenta en la arquitectura.

5.2.1. Fiabilidad

Fiabilidad - Disponibilidad

El atributo de calidad denominado disponibilidad, refiere a la capacidad de que el sistema esté total o parcialmente operativo para que el usuario pueda llevar a cabo las funcionalidades cada vez que lo requiera [24].

La disponibilidad juega un rol importante en la aplicación debido a que esta podrá ser la herramienta de trabajo para algunos de los usuarios. Es por dicha razón, que la no disponibilidad de la aplicación puede significar que el mismo no pueda desempeñar su labor. Para esto el sistema se hospedará en un servidor interno de la empresa. Este servidor, según los datos provistos, estuvo fuera de servicio por cuatro horas en el último año y 6 en el año anterior. Esto nos asegura que el servidor está funcional más de un 99% del tiempo. Además, se trazó un plan de contingencia en conjunto con el Administrador de Redes, de tal manera que se tenga un servidor secundario pronto para ser puesto en funcionamiento en caso de indisponibilidad del principal.

5.2.2. Usabilidad

La usabilidad es de los principales requerimientos no funcionales que se deben favorecer en el producto. Dado que los usuarios utilizarán la aplicación estando en la calle, la misma debe poder ser utilizada en esas condiciones sin presentarle un inconveniente al mismo.

Según la definición dada por D. Mordecki [25], "(...) *La usabilidad es la disciplina que tiene como objetivo reducir al mínimo las dificultades de uso inherentes a una herramienta informática, analizando la forma en que los usuarios utilizan las aplicaciones y sitios web con el objetivo de detectar los problemas que se les presentan y proponer alternativas para solucionarlos, de modo que la interacción de dichos usuarios con las aplicaciones y sitios web sea sencilla, agradable y productiva*".

Logrando que la usabilidad del sistema sea satisfactoria para los usuarios y respete los parámetros planteados para los distintos sub atributos de calidad, se obtienen ciertos beneficios que ayudarán al futuro de la aplicación.

- **Usuarios satisfechos:** Es el resultado de que los usuarios puedan conseguir sus objetivos con el mínimo esfuerzo posible.
- **Usuarios más fieles:** la facilidad de uso provoca que los usuarios deseen volver a utilizar la plataforma y de seguir indagando en sus funcionalidades.
- **Menor costo de soporte:** una aplicación fácil de usar genera menos problemas a los usuarios por lo cual se reduce la necesidad de soporte y ayuda.

- **Menor costo de mantenimiento:** los problemas de usabilidad surgen de quejas de los usuarios, generando modificaciones, por lo cual es mejor realizar las aplicaciones lo más usables posibles desde una primera instancia.

Usabilidad – Facilidad de Operación

Para asegurar la operatividad en todas las operaciones, se desarrollaron todas las funcionalidades de la aplicación de manera que no se necesite más de 3 *clicks* para acceder a ellas. La implementación de un menú lateral fue una herramienta de gran ayuda para poder cumplir con esto.

Usabilidad – Estética de la interfaz de usuario

Para cumplir con este atributo de usabilidad del sistema, se realizaron validaciones y revisiones con expertos para asegurar la misma, mejorando la UI. Estas llevaron a varios cambios en la aplicación. Se pueden ver más detalles de esto en el capítulo de [Aseguramiento de calidad](#).

Otra medida realizada para asegurar que la aplicación sea usable para todos los usuarios, fue el estudio de visión cromática. Para poder ayudar a diferenciar distinta información o estados de la aplicación, se resolvió jugar con los colores Azul, Rojo, Blanco y Negro, dado que en todos los casos (a excepción de la Acromatopsia) no habría problemas de distinguir entre dichos colores. Este análisis se puede ver en mayor profundidad en el [Anexo 13 - del análisis de problemas de visión cromática](#).

Usabilidad – Facilidad de Aprendizaje

Para cumplir con este atributo, no solo se minimizó la cantidad de *clicks* necesarios para acceder a las distintas funcionalidades, sino que también se utilizó una interfaz intuitiva y sencilla de comprender.

Para poder validar esto, se realizó una prueba de entrenamiento y usabilidad de la aplicación con un grupo reducido de futuros usuarios de la misma. Gracias a esto se pudo comprobar que con un pequeño entrenamiento de 13 minutos promedio por funcionalidad, sería suficiente para poder aprender a utilizar cualquier funcionalidad del sistema. Este estudio se puede ver en el [Anexo 14 - Estudio Facilidad Aprendizaje](#).

5.2.3. Eficiencia

La eficiencia es el tiempo requerido del sistema para responder a determinado evento [24].

Eficiencia – Comportamiento temporal

La aplicación fue desarrollada para poder procesar de forma asincrónica las consultas que manejan mayor cantidad de datos. Por ende, las más factibles a demorar más de 5 segundos son *obtener las paradas*, *obtener datos de recorridos* y *obtener próximos arribos a las paradas*. De esta forma, el usuario puede utilizar la aplicación sin percibir que existen procesos corriendo al mismo momento.

Las consultas previamente descritas se realizan mediante el uso de múltiples hilos corriendo en la aplicación. Se permite navegar por distintas pantallas de la misma, mientras que, de forma concurrente, se accede a la información procedente del servidor y se carga en pantalla.

En los ejemplos anteriores, el usuario es notificado que la información está cargada y lista para visualizarse, ya sea con un cartel de alerta o con un sonido y vibración.

Para poder asegurar que todas las funcionalidades del sistema demoran menos de 5 segundos en realizarse, se realizó una prueba de eficiencia sobre la totalidad de los servicios, en donde se pudo establecer que el tiempo de respuesta promedio de los servicios fue de 0.68 segundos y el tiempo máximo fue de 0.89 segundos. Valores inferiores a los 5 segundos planteados.

En el siguiente gráfico se puede ver a modo de resumen los tiempos de respuesta de los distintos servicios a lo largo de la prueba de 10 minutos de duración. El total de peticiones fue de 14.040, todas menores a 1 segundo. El eje horizontal representa el tiempo transcurrido y el eje vertical representa el tiempo de respuesta a las peticiones.

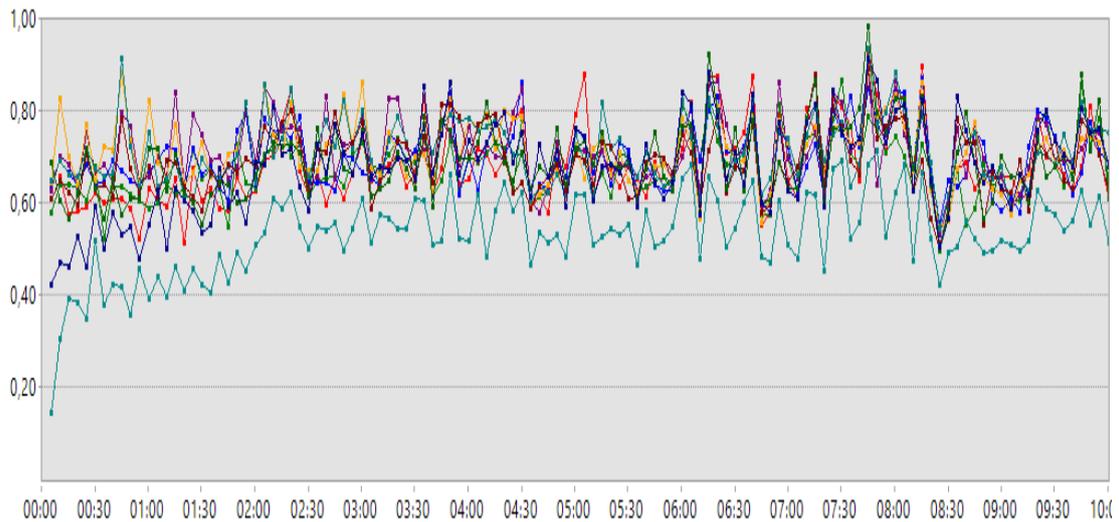


Gráfico 5-1 Tiempos de respuesta de servicios

La prueba de eficiencia completa se encuentra en el [Anexo 15 - Prueba de carga y eficiencia](#).

Eficiencia – Capacidad

Para poder asegurar que el sistema soporte 5000 usuarios, se realizaron pruebas de carga con una cantidad determinada de usuarios accediendo de forma concurrente. El equipo se basó en un cálculo fundado en la teoría de M. Baron para poder obtener este valor, de la cual se pueden ver mayor detalle en [26], y en [27] para ver detalles de su implementación.

Se definen a continuación las variables necesarias para el cálculo:

Variable	Significado
C	Número de usuarios concurrentes.
n	Número de usuarios totales.
L	Tiempo promedio de uso de la aplicación para todos los usuarios.
T	Intervalo de tiempo durante el cual estamos realizando la medición.

Tabla 5-1 – Significado variables teoría M. Baron.

Número de usuarios totales (n): La cantidad de usuarios que se logró alcanzar con el desarrollo fue de 5000, lo cual incluye a todos los inspectores, conductores, guardas, conductores cobradores y largadores. Como conclusión: **n = 5000 usuarios**.

Tiempo promedio de uso de la aplicación para todos los usuarios (L): Teniendo como base los tiempos que les llevó a usuarios algunas de las funcionalidades del sistema, el cual se puede apreciar en el [Anexo 14 - Estudio Facilidad Aprendizaje](#), se realizó una estimación del promedio de utilización de la aplicación por cada vez que se utiliza la misma, lo cual arrojó como resultado: **L = 9 minutos**.

Intervalo de tiempo durante el cual estamos realizando la medición (T): En promedio la jornada laboral de los empleados es de 8 horas, y en general se trabaja de lunes a sábado (6 días). Por lo que, haciendo cuentas, tendríamos: 6 días x 8 horas x 60 minutos = 2880, entonces: **T = 2880 minutos**.

Finalmente calculamos el número de usuarios concurrentes con los cuales deberíamos probar nuestros servicios, esto se hizo siguiendo la presente fórmula:

$$C = (n * L) / T$$

$$C = (5000 \text{ usuarios} * 9 \text{ minutos}) / 2880 \text{ minutos} = 16$$

Se puede ver en la ilustración 5-2 el resumen de la corrida de la prueba de carga, donde el sistema soportó durante 10 minutos a 16 usuarios concurrentes haciendo peticiones, donde ninguna de las mismas resultó fallida.

Overall Results	
Max User Load	16
Tests/Sec	2,34
Tests Failed	0
Avg. Test Time (sec)	6,79
Transactions/Sec	0
Avg. Transaction Time (sec)	0
Pages/Sec	23,6
Avg. Page Time (sec)	0,68
Requests/Sec	23,6
Requests Failed	0
Requests Cached Percentage	0
Avg. Response Time (sec)	0,68
Avg. Content Length (bytes)	112

Ilustración 5-2 Resultados de ejecución de pruebas de carga.

La prueba de carga completa se encuentra en el [Anexo 15 - Prueba de carga y eficiencia.](#)

5.2.4. Mantenibilidad

Mantenibilidad - Modificabilidad

La modificabilidad de un sistema apunta al costo e impacto que lleva aplicar cambios en la plataforma. [24]

Uno de los desafíos más grandes para el proyecto fue diseñar una arquitectura que se acople al cliente y pueda ser modificable por ellos. Para lograr esto, se realizó una investigación sobre cómo es el desarrollo en .NET en la empresa, qué arquitectura manejan y sus estándares.

En base a los resultados de la investigación, se diseñó una arquitectura que contemple las necesidades del proyecto y el estándar del cliente.

Para lograrlo, se diseñaron capas lógicas con responsabilidades bien asignadas, lo que facilita determinar el impacto al momento de realizar modificaciones para un nuevo negocio, e incorporar nuevos componentes en caso que sea necesario.

Asimismo, cada módulo que representa a usuarios distintos están independizados uno del otro y, de esta forma, un cambio en la funcionalidad de un módulo no afecta a otro.

Dada la existencia de funcionalidades genéricas útiles para todos los usuarios, existe un módulo general, el cual la aplicación siempre accede para obtener estas funcionalidades

El proyecto no contó con un caso en el que 2 usuarios compartan funcionalidades, pero de ser así, primero se evalúa la posibilidad de que sea común a todos, y de no serlo se deberá crear un módulo específico para esta funcionalidad. De esta forma, podrá ser identificada fácilmente en el sistema. Esto último favorece al atributo de calidad en cuestión, pero afecta la eficiencia. Dada la naturaleza de los requerimientos obtenidos, esta característica es poco común por lo que no tendrá un gran impacto en la eficiencia.

El manejo de error y la seguridad son transversales a todo el sistema, por lo que se encuentran en un módulo aparte, accesible por todos.

5.2.5. Seguridad

Seguridad - Confidencialidad

Para cumplir con este atributo de calidad, el sistema fue diseñado en función de módulos independientes, los cuales agrupan funcionalidades comunes a distintos roles.

Los roles son los que se usan actualmente en la empresa, con la salvedad de que, para el proyecto realizado, solo se llegó a alcanzar algunos de ellos (Inspectores, guardas y conductores y largadores). De todos modos, hay que tener en cuenta que el sistema se va a extender a nuevos roles de usuarios en el futuro.

Al momento del inicio de sesión de un usuario, el *backend* verifica el rol del mismo en la base de datos y solamente le ofrece a éste las funcionalidades para las cuales dicho rol está habilitado a usar.

Seguridad – Autenticidad

Para cumplir con este atributo, se implementó el uso de sesión de usuario. Para poder garantizar la identidad de quien está operando, se debió asegurar que no pasen largos periodos sin utilizar la aplicación, sin que tenga que re identificarse en el sistema, evitando así el posible uso indebido por parte de un tercero.

Hay 2 motivos por los cuales se le puede pedir a un usuario que se vuelva a identificar:

- Si tuvo un tiempo de inactividad mayor a 30 minutos (sesión expirada por inactividad).
- Si desde que se identificó en el sistema pasaron 8 horas (duración de una jornada laboral), independientemente si utilizo alguna funcionalidad de la aplicación o no. De esta forma, obliga a los usuarios a identificarse al menos 1 vez por día al comenzar la jornada.

Para la implementación de la sesión de usuario, se utilizaron *tokens* autogenerados que identifican a un usuario específico con una sesión.

Seguridad – Integridad

Para cumplir con este atributo, se tomaron dos medidas distintas. En primer lugar, se decidió cifrar la contraseña de los usuarios tanto en la base de datos como del lado de la aplicación móvil al momento de ser digitada, viajando hasta el *backend* de forma encriptada. Para el encriptado se utilizó AES como algoritmo de encriptación simétrica, utilizando una única clave de encriptación de 128 bits de tamaño.

Por otro lado, se utilizó un canal de comunicación seguro para todas las comunicaciones realizadas entre la aplicación móvil y el *backend*, utilizando TLS/SSL y un certificado de

seguridad. De esta forma, la información viaja encriptada y no puede ser interceptada por terceros.

Durante la implementación del encriptado del canal, se investigó sobre distintos tipos de certificados, descubriendo que los que se podían crear por defecto utilizando la herramienta de IIS, son del tipo SHA-1, función hash que ya fue quebrada previamente [28] y además va a dejar de ser considerada segura por Google en los próximos meses [29]. Razón por la que se trabajó en crear e instalar en el IIS un certificado más seguro del tipo SHA-2, mismo tipo de certificado utilizado por entidades bancarias, por ejemplo. El mismo aún no fue quebrado y brindara mayor seguridad a las comunicaciones del sistema. Se puede ver más detalle en la siguiente ilustración.

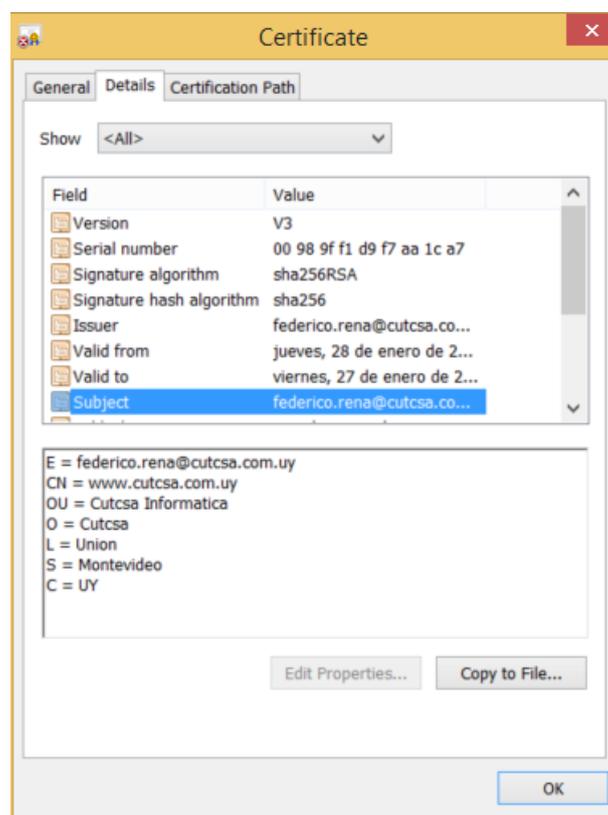


Ilustración 5-3 Certificado SHA-2 creado por el equipo

Para poder comprobar la correcta implementación del encriptado del canal de comunicación, se realizaron pruebas en conjunto con el administrador de redes de la empresa, las cuales se pueden ver en el [capítulo de calidad en la sección ejecución y resultados](#).

5.3. Descripción de la arquitectura

5.3.1. Patrones

Observer: Dentro de la aplicación móvil, se utilizó dicho patrón para poder actualizar el estado de una actividad cuando la misma realiza otras acciones en hilos concurrentes al principal.

Singleton: Tanto en la aplicación móvil como en el *backend*, este patrón es utilizado para clases que encapsulan valores necesarios en memoria y solamente se necesita una única instancia de las mismas.

Fachada: Este patrón se utiliza para el menú de la aplicación, ya que todas las opciones del mismo tienen como punto de acceso el menú lateral.

Command: Este patrón se utiliza para llamar a los servicios desde las actividades y, de esta forma, se puede generalizar los llamados, ya que cuando se llama a los servicios no se conoce qué operación está consultando ni el resultado de la misma. La actividad realiza el llamado, la aplicación los procesa, sin saber quién ni qué está demandando un servicio, lo ejecuta y le retorna la respuesta a la actividad.

Abstract: Las actividades que usan el menú y el mapa lo hacen heredando de una clase genérica de mapa y de menú. De esta forma, se pueden redefinir o extender funcionalidades de estas clases para utilizar en casos particulares.

5.3.2. Vistas Arquitectónicas

Vistas lógicas

En esta arquitectura se propone un diseño basado en el estilo de capas lógicas las cuales proveen diferentes niveles de abstracción.

Con esto se logra que cada componente cuente con un conjunto finito de funcionalidades y responsabilidades.

La selección de la arquitectura se basó en las buenas prácticas de diseño favoreciendo la mantenibilidad y modificabilidad del sistema.

En esta sección, el principal lineamiento con la arquitectura del cliente son las entidades del negocio y las excepciones como capas transversales, ya que las soluciones de los proyectos del cliente fueron realizadas con la misma estructura de capas.

La separación en capas lógicas permite minimizar el acoplamiento entre los componentes dado que cada uno de ellos se relaciona con otro, con el mismo o menor nivel de abstracción.

Los diferentes componentes de las capas exponen sus funcionalidades a través de interfaces bien definidas. Con esto se logra exponer únicamente las funcionalidades necesarias y se ocultan los mecanismos internos de cada componente, favoreciendo el reúso de estos y la modificabilidad del sistema. A su vez, esta agrupación en los diferentes niveles de abstracción permite que, en un futuro, dichas capas puedan ser distribuidas físicamente modificando únicamente los mecanismos de comunicación.

El lado negativo del diseño propuesto es que los cambios de comportamiento generados en los diferentes componentes pueden generar un impacto en las otras capas dada la propagación de dicho cambio.

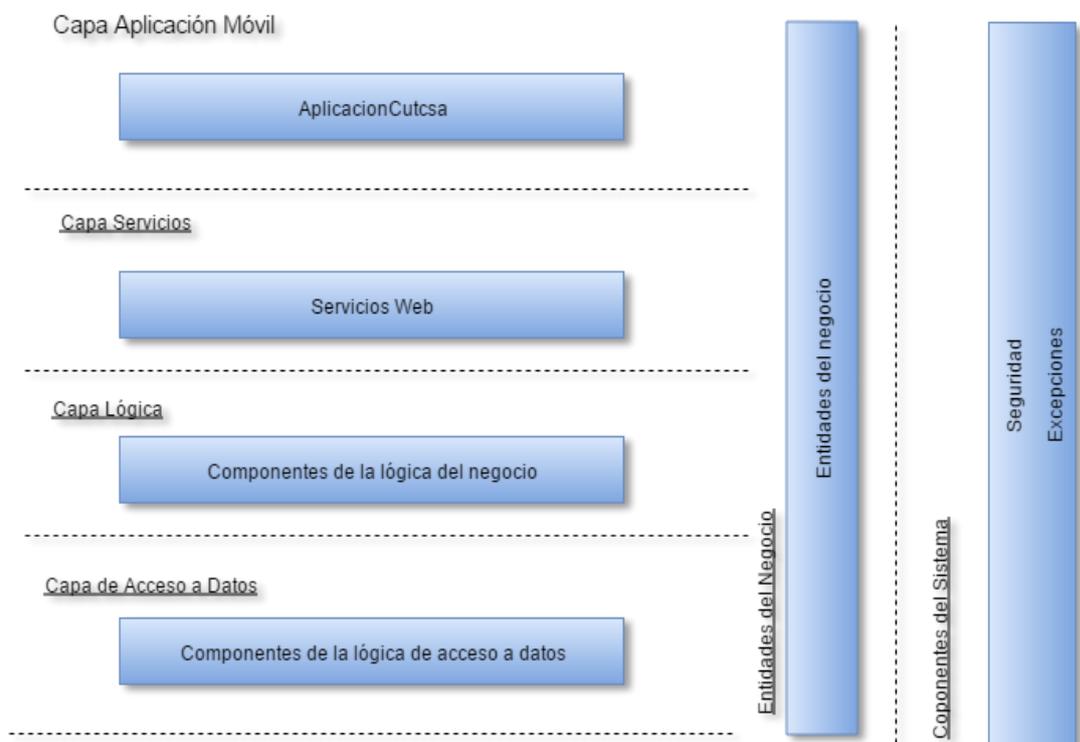


Ilustración 5-4 Vista de capas Lógicas

Como se comentó anteriormente, se pueden apreciar claramente las diferentes capas de la aplicación, las cuales son:

- Capa Aplicación Móvil: Dentro de esta capa se encuentran los componentes dedicados a la interfaz gráfica de la aplicación, así como el manejo de la comunicación con los servicios.
- Capa Servicios: Esta capa es la encargada de publicar los servicios que se utilizarán en la aplicación .web, api, .net.
- Capa Lógica: Dentro de esta capa se encuentra la lógica de la aplicación, así como también las diferentes reglas del negocio.
- Capa de Acceso a Datos: Esta capa es la encargada de proveer de datos a la aplicación.
- Capa de Intercambio de Datos: La responsabilidad de los componentes de esta capa es la comunicación entre la base de datos de C.U.T.C.S.A. y la base de datos que se utilizará en la aplicación.
- Capas transversales:
 - Entidades de negocio: provee a la aplicación de los objetos necesarios para la representación correcta de las entidades del dominio.
 - Componentes del sistema: Dentro del sistema existen diferentes requerimientos y funcionalidades las cuales se han agrupado en componentes que resuelven dichos casos. Entre ellas se encuentra las excepciones y seguridad.

Vista de despliegue

La arquitectura de despliegue fue sugerida por el cliente, la cual fue pensada en la seguridad del sistema. Una de las características fundamentales de la misma es el hecho que la solución que se desarrolló, está desacoplada de los sistemas existentes. Si bien la solución depende de estos para el acceso a datos, el desarrollo se realizó de forma independiente teniendo en cuenta los atributos de calidad mencionados anteriormente. Es decir que, si bien existieron restricciones, también hubo libertad de utilizar cualquier otra tecnología que fuese necesaria.

La otra característica fundamental de esta arquitectura es que fue pensada en la seguridad de los datos. Los datos en la base de datos de C.U.T.C.S.A. son muy sensibles y actualmente no se pueden acceder sin ser dentro de la red local. Mediante una nueva base de datos del proyecto, que solo tendrá información relacionada con la aplicación, se podrá proteger la base de datos principal de la empresa. Ya que se utilizarían solamente datos necesarios.

Para obtener los datos de la empresa, se sugirió la creación de un intermediador que cumplirá el rol de volcar los datos necesarios de una base a otra. Este solo podrá leer datos de la base de datos de C.U.T.C.S.A y guardar en la base de datos del proyecto. Esta operación se realiza por la red local. De esta forma, en caso de existir un ataque al sistema y en el caso que puedan acceder a la base de datos del proyecto, no tendrán

acceso a ningún otro nivel de datos más que en el que se encuentran. De todas formas, como se mencionó anteriormente, los datos más sensibles se encuentran cifrados y las comunicaciones son seguras gracias a la implementación de TLS/SSL.

En la ilustración 6.3 se aprecia el diagrama de despliegue en el cual se indica qué áreas fueron incluidas en el proyecto y cuáles fueron externas al mismo. A continuación, se detallan los elementos del diagrama ordenados por importancia.

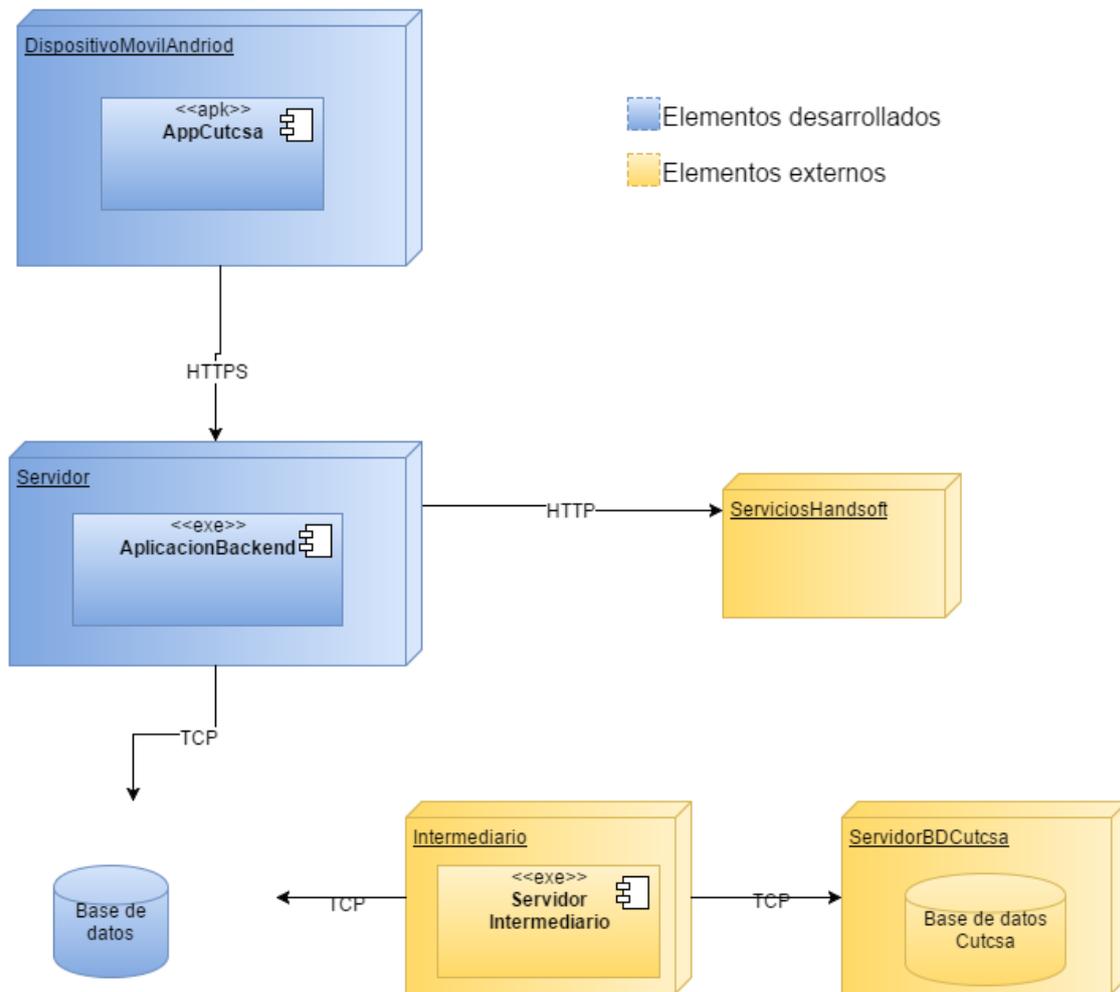


Ilustración 5-5 Diagrama de despliegue

Elemento	Responsabilidad
DispositivoMovilAndriod	Se encuentra la aplicación que utilizará el usuario. Este tendrá las funcionalidades relevadas.
ServidorBD	Contiene la base de datos del proyecto con la cual la aplicación obtendrá los datos. Solo tendrá los datos que van a usarse en la aplicación.
ServiciosHandsoft	Los servicios de Handsoft contienen la información que se utiliza en la aplicación relacionada a la posición de los coches y el tiempo de demora de un coche para trasladarse de un punto a otro.
Intermediario	Intermediario entre la base de datos de C.U.T.C.S.A. y la base de datos que accederá la aplicación <i>backend</i> , cuyo objetivo es mantener la seguridad de los datos de la primera y actualizar la base de datos del proyecto.
ServidorBDCutcsa	Este componente no se modificará, solo se obtendrán los datos necesarios del mismo para volcarlos luego en la base de datos del proyecto.

Tabla 5-2 Elementos del diagrama de despliegue

Vista de Componentes

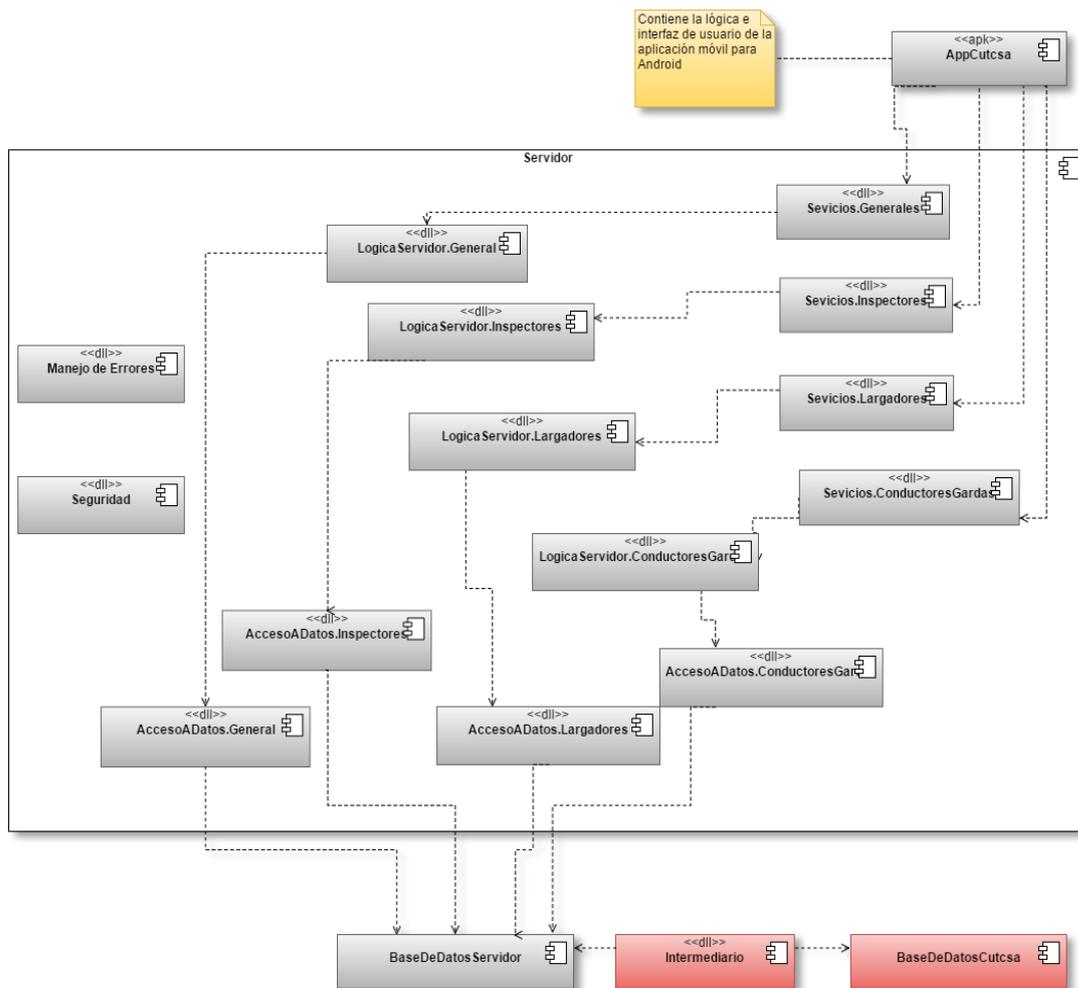


Ilustración 5-6 diagrama de Componentes

En la ilustración 5.6 se puede apreciar los distintos componentes del sistema. En rojo se encuentran aquellos que no fueron desarrollados por el equipo. Los componentes de Seguridad y Manejo de Errores están representados como si no tuviesen ninguna relación con el resto de los componentes, pero es así para simplificar el diagrama ya que de estos dependen todos los componentes del *backend*.

La estructura del componente servidor se diagramó de esta forma ya que en C.U.T.C.S.A. las aplicaciones .net están hechas con esta estructura. La decisión fue tomada para conservar la misma solución, ya que la aplicación va a ser mantenida eventualmente por la empresa.

Elemento	Responsabilidad
AppCutcsa	Aplicación que utilizará el usuario
Servicios.Generales	Brindar los servicios que se utilizarán en la aplicación para funcionalidades comunes a todos los módulos
Servicios.Largadores, Servicio.Inspectores, Servicios.ConductoresGuardas	Brindar los servicios que se utilizarán en la aplicación. Cada uno de estos componentes sirven únicamente para el módulo para el cual fueron creados.
LogicaServidor.General	Manejar la lógica y los datos que se utilizarán en la aplicación para las funcionalidades comunes a todos los módulos
LogicaServidor.Largadores, LogicaServidor.Inspectores, LogicaServidor.ConductoresGuardas	Manejar la lógica y los datos utilizarán en la aplicación. Cada uno de estos componentes sirven únicamente para el módulo para el cual fueron creados.
ManejoDeErrores	Realizar validaciones comunes y del manejo de excepciones del <i>backend</i>
Seguridad	Tiene la lógica correspondiente a la solución de seguridad y manejo de sesión.
AccesoADatos.Largadores, AccesoADatos.Inspectores, AccesoADatos.ConductoresGuardas	Traer y guardar datos de la base de datos para las funcionalidades específicas de cada módulo para el cual fueron creados.

Tabla 5-3 Elementos del diagrama de componentes

Diseño de la Aplicación Móvil

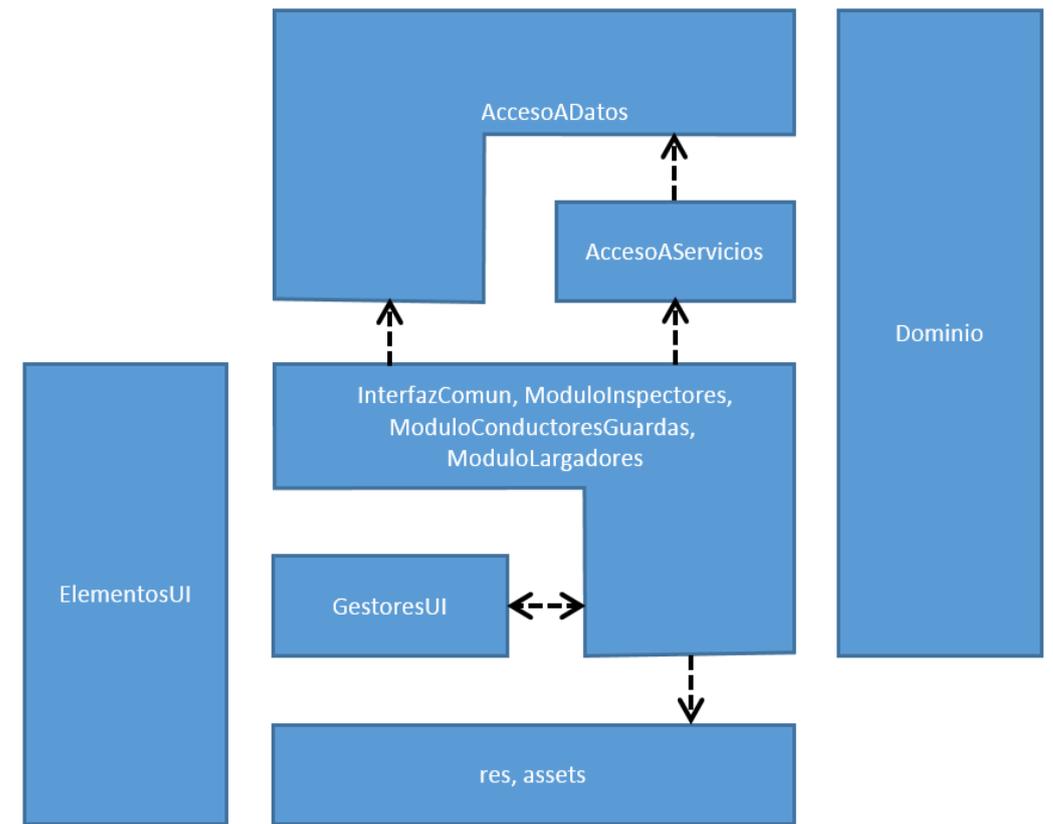


Ilustración 5-7 Diagrama de paquetes Aplicación Móvil

La aplicación móvil está dividida en 4 grandes paquetes con objetivos específicos: Interfaz, Dominio, AccesoAServicios y AccesoADatos, alguno de ellos con sub paquetes, así como paquetes de uso obligatorio para las aplicaciones Android como son Res y assets. A continuación, se detalla cada uno de ellos y sus responsabilidades.

Interfaz:

- **Activities:** Se encuentran dentro de los paquetes InterfazComun, ModuloConductoresGuardas, ModuloInspectores, ModuloLargadores. Son todas las actividades que se despliegan al usuario. Es un código que administra los componentes gráficos, como ser la carga de información en listas, acciones de botones, etc.
- **GestorComponentesDinamicos:** Se encarga de la creación de elementos de ciertas vistas que se despliegan al usuario, cuyos datos se definen de forma dinámica, y que a su vez se utiliza varias veces durante el ciclo de vida de la aplicación.
- **GestorListeners:** Gestiona acciones similares que realizan distintos componentes dentro de la interfaz de usuario, y que se engloban en métodos o interfaces reusables.

- **GestorLocalizacion:** Administra las acciones y obtención de datos referentes a la localización del dispositivo, a través del uso de GPS y otras redes.
- **GestorMensajes:** Gestiona los componentes que sirven de alerta al usuario (*popups*, *notifications*, etc.) de forma genérica y reusable según el tipo de mensaje que se desea mostrar al usuario (mensaje de aviso, mensaje de error, mensaje de carga, notificación). Dependiendo del tipo de *popup*, el mismo se cierra solo, o existe un método genérico dentro de las actividades (que se encuentran dentro del paquete *InterfazComun*) que se encarga de cerrarlo al realizar un llamado al método.
- **GestorMenu:** Gestiona los accesos a las opciones que se despliegan en el menú lateral. Cuando el usuario presiona sobre un elemento de la lista del menú lateral, el mismo se encuentra asociado a un número identificador, el cual es pasado como referencia a esta clase. La misma, a través de un *switch*, identifica dicho número y llama al método que se encarga de abrir la actividad para mostrarse al usuario.
- **ElementosUI:** Es un paquete que almacena todos los elementos creados en forma particular y a medida para distintos componentes gráficos (adaptadores de elementos de listas, imágenes, visualizador de *GIFs*, menú lateral, etc.).

Dominio:

- **Sistema:** Se encarga de almacenar al usuario, un *stack* de las actividades abiertas (y lógica de cuándo y cómo cerrarlas), y carga de información básica para la aplicación (dependiendo del usuario que inició sesión. Se explica más adelante).
- **Clases de Dominio:** Agrupación de datos para todos los elementos que pertenecen al dominio, para ser usado a lo largo de toda la aplicación.

AccesoAServicios:

Clases que se encargan del envío y obtención de información hacia y desde el servidor, y procesado para ser almacenado o mostrado al usuario. Dichas acciones se realizan en un *Thread* independiente al principal (para no sobrecargarlo ni trancar la interfaz gráfica). A la hora de devolver información a la interfaz de usuario (ya sea los datos obtenidos o procesados del servidor, o errores que pudieran surgir), se utiliza el *Listener* *IAccesoAServiciosCallback*, el cual permite a la clase de la interfaz que llamó al método del acceso a servicios, obtener la información o error para ser mostrada al usuario. El método *onFinalizar* de dicha interfaz, envía como resultado un objeto genérico (*Generic Type <T>*), que es procesado del lado de la *Activity* según el dato que se está esperando.

```
public interface IAccesoAServiciosCallback<T> {  
  
    void onFinalizar(T resultado);  
  
    void onError(String informacionError, Exception excepcionError);  
  
    void onActualizar();  
}
```

Ilustración 2 Implementación del listener IAccesoAServicioCallback

AccesoADatos:

Paquete que alberga la clase que se encarga de la gestión de la base de datos interna del dispositivo, para almacenar y acceder la información que se necesita a lo largo de todo el ciclo de vida de la aplicación.

Res, assets:

Almacena agrupaciones de componentes para *Activities* u otros componentes para ser mostrado al usuario (dentro de la carpeta *layout*), imágenes usadas dentro de la aplicación (dentro de la carpeta *drawable*), sonidos (dentro de la carpeta *raw*), *GIFs* (dentro de la carpeta *assets*), y distintos *strings* para mensajes, alertas, notificaciones, componentes, etc. (dentro de la carpeta *values*).

5.4. Decisiones generales de tecnologías utilizadas

Las tecnologías utilizadas en el proyecto fueron seleccionadas con el objetivo de que ayuden a cumplir con los objetivos del proyecto y se adapten mejor a las necesidades del equipo, por lo que se utilizaron ciertos para la elección de las mismas. Estos fueron:

- Restricciones impuestas por el cliente
- Facilidad de aprendizaje
- Conocimiento y experiencia del equipo en la tecnología
- Adecuación a los requisitos

5.4.1. Herramientas de desarrollo.

Para el desarrollo del *backend*, se utiliza C# en .net a través de Visual Studio, y para el desarrollo de la aplicación móvil, Java a través de Android Studio. Estas tecnologías son restricciones para el proyecto, por lo que el no uso de las mismas estaría incumpliendo con el acuerdo realizado con el cliente. El Android Studio se seleccionó dada la facilidad de aprendizaje que presenta la herramienta con respecto a las otras existentes como el Eclipse de Oracle. Esta conclusión se obtuvo luego de haber trabajado con ambas. También se encontró que es más sencilla de aprender ya que existen tutoriales muy completos sobre la misma.

Repositorio

Como repositorio se utiliza Git, y Bitbucket como cliente para la aplicación móvil y Team Foundation Server como repositorio del *backend*. Las decisiones de por qué se seleccionaron estas tecnologías se encuentran en el capítulo de [Gestión de la configuración](#).

Conexión con la base de datos

Para el manejo de datos se utiliza Entity Framework. Dentro de las alternativas que provee y de acuerdo a las necesidades del proyecto, se decide usar *Code First* debido a que los requerimientos, si bien varían para cada módulo, son estables y se ha invertido una cantidad de tiempo considerable para realizar la especificación de los mismos en cada evolución. Esta herramienta se utilizó dado el conocimiento que se tenía de la misma por parte de todos los integrantes del proyecto y también se adecua al requisito de desarrollar en .net.

Base de datos

Para la base de datos se utiliza SQL Server por la integración que existe, utilizando Entity Framework y .net en Visual Studio. Al igual que el caso anterior, esta herramienta se selecciona por el conocimiento de los integrantes en la misma y su adecuación con los requisitos impuestos.

Esta plataforma ofrece también varios atributos de seguridad que ayuda a proteger los datos. También cuenta con control de acceso para poder tener un mayor control sobre quién o qué puede acceder a la base.

La integridad de los datos es otro de los puntos fuertes de esta plataforma, SQL Server proporciona varios mecanismos para exigir la integridad de datos.

Comunicación

Para la comunicación entre la aplicación y el *backend* se utilizan servicios REST y HTTPS.

REST se seleccionó debido a la separación que permite entre la aplicación y el servidor, y también por la independencia de tecnologías y lenguajes. Estas 2 características de REST permiten completa independencia del *backend* y la aplicación. A su vez, a raíz de que el sistema seguirá siendo desarrollado por la empresa en el futuro, si estos deciden agregar una aplicación móvil con otro sistema operativo, o mismo el *backend* de la aplicación, se puede realizar sin tener impacto en otra área de la solución [30].

Los parámetros que maneja REST son *strings* pasados por parámetro al llamado del servicio en un orden particular o, en el caso que sea mucha información que se deberá comunicar, se utiliza JSON.

Para la creación de los servicios, se utilizó WCF por el conocimiento que tenía el equipo sobre la tecnología.

5.5. Conclusiones

5.5.1. Lecciones Generales

Cada tecnología, por más simple que parezca al principio, tiene su curva de aprendizaje e investigación, más aún cuando se integra con otro conjunto considerable de tecnologías.

Reestructurar y refactorizar el código de forma seguida fue fundamental, principalmente para la aplicación móvil ya que permitió ayudar a cumplir con el objetivo de mantenibilidad de la aplicación, debido a que el código luego de cada refactorización resultaba más entendible para el programador.

5.5.2. Lecciones Sobre tecnologías

Con respecto a la selección de las tecnologías, se eligió WCF simplemente por el conocimiento del equipo sobre esta tecnología sin haber realizado una investigación previa. Esto causó sobreesfuerzo en algunas áreas de desarrollo que no fueron previstos por no haber investigado si era la mejor opción para el proyecto. Hubo dos principales problemas. Uno fue encontrar cómo puede retornar el servicio un objeto JSON plano sin nada más en el cuerpo del mensaje. El otro problema fue su compatibilidad con SSL. Fue muy complicado adaptar ambas tecnologías dados los pocos ejemplos encontrados que mostraban ambas tecnologías trabajando en conjunto. Si tuviésemos la oportunidad de

pasar por la etapa de elección de tecnologías, lo que haríamos sería una mayor investigación sobre otras tecnologías .NET que nos permita realizar los mismos comportamientos que WCF para evaluar si podría resultar más útil el uso de otra tecnología, como en este caso podría haber sido Web Api, que cuenta con soluciones de seguridad ya desarrolladas.

Entity Framework y SQL Server fueron muy útiles para adaptarse a la restricción de la tecnología .NET. El conocimiento previo que se tenía sobre estas tecnologías ayudó a facilitar la tarea de diseño y desarrollo.

6. Proceso de desarrollo

En este capítulo muestra y explica el proceso de desarrollo utilizado, las principales decisiones tomadas a lo largo del mismo, cómo se trabajó, y mostrar el alcance logrado.

6.1. Características del equipo

Es importante describir en primer lugar los conocimientos de los distintos integrantes del equipo previos a la realización del proyecto.

A nivel académico, todos los integrantes tienen experiencia de desarrollo con diversas tecnologías, siendo las principales: Java y .NET, y sólo un integrante también con Android.

A nivel profesional, el integrante con experiencia académica en Android también posee 1 año de experiencia profesional en dicha tecnología y además 2 años más con tecnologías .NET. De los integrantes restantes, uno no tiene experiencia profesional en desarrollo, y el restante tiene 3 años de experiencia en desarrollo en Javascript y Netsuite.

Por lo que se puede denotar que no es un equipo con conocimientos y experiencias homogéneas, factores que se tuvieron en cuenta para crear el plan de desarrollo y que tuvieron un gran impacto en la elección de las tecnologías utilizadas, lo cual se puede ver con más profundidad en el [capítulo de Arquitectura](#).

6.2. Responsabilidades

Al margen de los roles ya definidos, que se pueden ver en la [Introducción](#), entre otras, se definió que todos los miembros del equipo fueran desarrolladores. Se decidió asignar responsabilidades dentro del equipo referentes a la etapa de desarrollo. A continuación, en la ilustración 7.1 se detallan dichas asignaciones:

Referente	Nombre
Referente móvil	Diego Barcia
Referente <i>backend</i>	Juan Pedro Trecca
Referente seguridad	Andrés Amegeiras

Tabla 6-1 Responsabilidades de los integrantes

En el caso del **Referente móvil**, se asignó a Diego Barcia como tal, ya que es el único integrante en contar con conocimiento académico y profesional en desarrollo móvil. Su labor consistió en desarrollar la totalidad del *frontend* del sistema, además de la implementación de la conexión del mismo con el *backend*. Si bien la principal tecnología utilizada fue Android, el desarrollo incluyó también interacción con otras como ser: NFC, códigos QR y diversas API's.

Para el caso del **Referente del backend**, se asignó a Juan Pedro Trecca como tal debido a su experiencia profesional en desarrollo y a su afinidad a las tecnologías .NET. A lo largo del proceso de desarrollo, se encargó de implementar la mayor parte de los requerimientos y servicios asociados. Algunas de las tecnologías utilizadas fueron C#, WCF y Entity Framework entre otras.

Para el caso del **Referente en seguridad**, en primer lugar, cabe destacar que la asignación de un integrante a dicha área no estaba en los planes del equipo, hasta que esto fue fuertemente sugerido por un docente de la Universidad durante una revisión. Debido a esto, y considerando que varios de los requerimientos no funcionales están relacionados a esta área, fue que se decidió aceptar la sugerencia. Ninguno de los miembros del equipo tiene experiencia previa en seguridad, por lo que se designó a Andrés Amegeiras como referente, debido a que los demás integrantes ya contaban con otras asignaciones. Durante el desarrollo, se encargó de investigar e implementar todo lo referente a la seguridad del sistema. Algunas de las tecnologías utilizadas fueron TLS/SSL, SHA, AES, entre otras.

Dado el tamaño reducido del equipo, no se creyó necesario definir equipos de desarrollo por áreas, aunque, de todos modos, informalmente se terminó generando una división tal que los integrantes referentes al *backend* y a la seguridad trabajaron en la implementación de los requerimientos del lado del *backend* y por otro lado, el referente móvil trabajó en la implementación de todos los requerimientos del lado del *frontend*.

6.3. Estrategia de desarrollo

En esta sección se detalla la estrategia de desarrollo adoptada y cómo fue evolucionando a lo largo del proyecto.

En la siguiente ilustración se muestra una línea de tiempo en la que se presenta la estrategia de desarrollo para poder tener una referencia temporal de cómo se llevó a cabo:

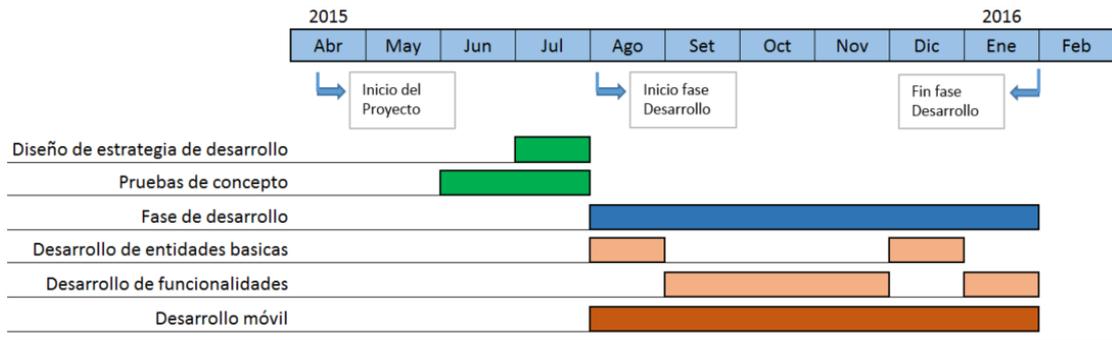


Ilustración 6-1 Línea de tiempo del desarrollo

Previo a comenzar la fase de desarrollo, el equipo diseñó una estrategia para poder llevar a cabo esta fase de forma eficiente y ordenada. La misma se diseñó entre todos los integrantes del equipo luego de evaluar las distintas opciones posibles.

Independientemente de la decisión de asignar un referente para la aplicación móvil y otro para el *backend*, inicialmente el plan fue que todos los integrantes desarrollaran parte de ambas cosas. Razón por la cual, 2 de los integrantes realizaron un entrenamiento Android diseñado por el referente del equipo de esa tecnología, para poder adquirir conocimientos previos a comenzar el desarrollo.

Luego de haber realizado ya la mitad de este entrenamiento y aun previo a comenzar el desarrollo, hubo un cambio en la estrategia de desarrollo. Debido a atrasos en otras tareas referentes a Ingeniería de requerimientos, y además re evaluando los conocimientos y situación del equipo, se decidió que iba a ser más efectivo dividir el desarrollo, asignando a los 2 integrantes sin experiencia Android a implementar el *backend*, dejando el *frontend* para el integrante referente de Android. Con esta decisión, se dejó abierta la posibilidad de aumentar recursos en el *frontend* en caso de ser necesario. Debido a esto, se decidió poner en pausa indefinida el entrenamiento Android.

Durante la fase de requerimientos y diseño, el equipo además realizó algunas pruebas de concepto para la aplicación Android, en las cuales se investigó y probó con ejemplos distintas tecnologías que iban a ser utilizadas durante el desarrollo. Entre ellas, la comunicación mediante NFC entre un dispositivo móvil y una tarjeta puente, lectura de códigos QR para poder identificar paradas de buses, Google places API referente a manejo de Google *Maps* y finalmente autocompletado de direcciones. De esta forma se aprovechó a adoptar tecnologías, y reducir el impacto y riesgo de utilización de las mismas durante el desarrollo.

Una vez comenzada la fase de desarrollo, lo primero que se hizo fue acordar los estándares de codificación para cada uno de los lenguajes utilizados, Java y C#. Luego, se procedió a comenzar con el desarrollo del dominio y las entidades básicas del mismo

por parte de los 2 integrantes asignados al *backend*, lo cual comprendió las altas, bajas y modificaciones. En paralelo, el integrante restante comenzó a desarrollar también las entidades básicas del *frontend*.

Luego, se continuó con la implementación de las funcionalidades tanto a nivel de *frontend* como de *backend*, además de trabajar en la comunicación y el formato de los datos a enviarse entre una parte y la otra.

Con la estrategia adoptada, se buscó que en el *backend* el desarrollo fuese vertical, es decir, que cada integrante, al momento de implementar una funcionalidad, tuvo que hacerlo partiendo desde la capa de presentación de los servicios hasta la capa de acceso a datos. Esto implicó que cada Desarrollador debiera enfrentarse con todas las tecnologías manejadas a nivel de *backend*.

Al comenzar la tercera evolución, y la adición a la solución de los nuevos roles, el equipo se dedicó a agregar estas nuevas entidades básicas al dominio, para luego continuar implementando funcionalidades, en este caso de la última evolución cursada.

A lo largo del desarrollo se plantearon varios retos, entre ellos, la utilización y dependencia de servicios externos al cliente, lo cual generó demoras y errores en la implementación en varias ocasiones. Por otro lado, la implementación de la seguridad de la aplicación demandó una dedicación de forma exclusiva a dicha actividad por parte del encargado de la misma y donde, para su implementación, tuvo que trabajar tanto en el *frontend* como en el *backend*.

Durante el desarrollo, la utilización de los referentes de área fue de gran importancia al momento de plantearse dificultades. En primer lugar, estas eran planteadas al referente del área involucrada, quien, en muchos casos, supo cómo resolverla, sin necesidad de haber desperdiciado recursos para la investigación de la solución al problema.

6.4. Alcance desarrollado

Se muestra a continuación el detalle de desarrollo alcanzado para cada uno de los requerimientos estipulados dentro del alcance del proyecto, los cuales se pueden ver en el [capítulo de ingeniería de requerimientos](#).

Id.	Requerimiento	Desarrollado
G-RF1	Log In	100%
G-RF2	Log Out	100%
G-RF3	Mostrar Menú	100%

G-RF4	Reportar un error en la aplicación	100%
G-RF5	Recepción de notificaciones	100%
G-RF6	Gestión de notificaciones	100%
G-RF7	Visualización de noticias	100%
C2-RF1	Ver información de paradas	100%
C2-RF2	Visualizar el recorrido de cierta línea en un mapa	100%
C2-RF3	Mostrar arribo de líneas a la parada	100%
C2-RF4	Ubicar coche en un mapa	100%
C2-RF5	Ver ruta entre un origen y un destino	0%
C2-RF6	Ver información de puntos de interés	100%
C2-RF7	Control del servicio	95%
C2-RF8	Mostrar horario planificado de una línea	100%
C3-RF1	Consultar servicios asignados	100%
C4-RF1	Modificar servicios	100%
C4-RF2	Alta servicio	100%
C4-RF2	Baja servicio	100%

Tabla 6-2 Porcentaje de avance de requerimientos

6.5. Conclusiones

Creemos que fue muy importante la utilización de una estrategia de desarrollo, ya que de esa forma se optimizaron los tiempos de desarrollo y las capacidades de cada integrante del equipo, sobre todo teniendo en cuenta que se trató de un equipo heterogéneo, donde la experiencia y conocimiento de los integrantes fueron dispares.

Finalmente, respecto al alcance logrado, se denota que se logró implementar en su totalidad 17 de los 19 requerimientos planteados para el proyecto académico.

De los 2 requerimientos restantes donde esto no se logró, uno fue el C2-RF7 Control de servicio, el cual fue implementado en un 95%, siendo el 5% restante debido a una dependencia externa, en la que el cliente no pudo resolver con el organismo regulador (I.M.), ver el [plan de proyecto](#) para más información. Por dicha razón, no se tuvo toda la información necesaria para concluir el requerimiento. De todos modos, se dejó implementada la estructura necesaria para que, cuando se obtenga la información faltante, la tarea de concluir el requerimiento sea trivial.

El otro requerimiento que no se pudo concluir, fue el C2-RF5 Ver ruta entre un origen y un destino, el cual no fue implementado, debido a su complejidad y al poco valor que

este aportaba al cliente, ya que actualmente existe varias aplicaciones alternativas que implementan lo mismo. A su vez, se puede ver en el [Anexo Encuesta Inspectores](#) que fue uno de los requerimientos de menor prioridad para los usuarios en una encuesta realizada a los mismos.

7. Gestión del proyecto

La gestión tuvo un peso importante en el proyecto, siendo la segunda actividad con mayor esfuerzo luego del desarrollo. En la misma, se decidió tener una fluida comunicación con el cliente, para que conozca en todo momento el avance del proyecto, y tomar medidas correctivas en los casos que existieron cambios.

A su vez, con una fuerte dedicación en el análisis y toma de métricas, se buscó, ir mejorando, evolución a evolución, las estimaciones realizadas, lo cual efectivamente se logró.

A lo largo del capítulo, se detallarán las actividades realizadas para la gestión del proyecto, su planificación y seguimiento.

7.1. Estrategias de gestión y planificación temporal de tareas

Como estrategia, se buscó un enfoque predictivo de gestión de proyectos, basado en experiencias anteriores, y posteriormente, análisis que se realizaron sobre la marcha, con el objetivo de mejorar la precisión de la estimación.

A su vez, se decidió realizar un plan de proyecto evolutivo, es decir, a medida que se avanzaba en el cronograma, se realizaban actualizaciones al plan. Cuando se finalizó cada revisión, teniendo en cuenta las correcciones y sugerencias del revisor, se generó una nueva versión actualizada de dicho plan para ser entregada al *Sponsor* y que el mismo lo valide. Los datos que se actualizaron fueron descripciones de avances, tareas realizadas y análisis.

Para la planificación del proyecto, primariamente se tuvieron en cuenta muchas restricciones y fortalezas por parte del equipo y del proyecto en sí. Las más importantes fueron:

- Producto innovador: para C.U.T.C.S.A., una aplicación móvil que apoye la gestión y las tareas de los empleados internos constituyó un producto que no existía en el contexto local y, por ende, no había una base con la cual apoyarse para definir como solución a los problemas y oportunidades de mejora detectados.
- Tecnologías poco conocidas por el equipo: las tecnologías definidas para el desarrollo del proyecto no eran conocidas en su totalidad por todos los integrantes del equipo. Esto obligó a destinar tiempo para su investigación.
- Empleado de la empresa como integrante del equipo: tener a uno de los miembros del equipo como empleado de C.U.T.C.S.A. fue importante para agilizar la comunicación entre las distintas partes interesadas.

Todas estas características afectaron la precisión para la definición del cronograma y su estimación.

La ilustración 7.1 muestra una línea de tiempo con los principales hitos del proyecto.



Ilustración 7-1 Línea de tiempo general del proyecto

Para el desarrollo de la solución se decidió realizar 3 evoluciones en total. En primer lugar, una evolución con el objetivo de armar la base de la aplicación móvil y desarrollo del *backend*, junto con algunos requerimientos comunes para todos los usuarios. Luego una segunda evolución, para desarrollar los requerimientos particulares para los inspectores. Y finalmente, una tercera evolución para desarrollar funcionalidades para los conductores, guardas, conductores cobradores, largadores, y algunas funcionalidades comunes.

En particular, se decidió realizar las dos primeras evoluciones concomitantemente. Por un lado, la primera evolución por una restricción técnica, ya que era necesaria una arquitectura base en la que el resto de las funcionalidades se basarían. Y, por otro lado, a raíz de que ya existía un pequeño prototipo para las funcionalidades de los inspectores, se decidió realizar la segunda evolución también, conectando de esta manera ambas partes del sistema (*backend* y *frontend*). Pero dicho desarrollo no generó problemas de dependencia, ya que mientras se realizó la base de la arquitectura del *backend*, se fue realizando el diseño de las actividades y vistas (*Activities* y *Views*, del término Android) de las funcionalidades para los inspectores, siempre teniendo en cuenta la descripción de la arquitectura definida.

Durante la ejecución, se prestó una atención especial a la disponibilidad de los distintos interesados del proyecto, anticipando posibles casos que estuvieron ocupados, y adelantando reuniones lo más tempranamente posible.

Para las asignaciones de tareas más específicas, se utilizaron herramientas que ayudaron a organizar el equipo y que todos estén al tanto de las tareas que se les asignó, fechas límites, y descripciones de las mismas. De esta forma, también se pudo realizar un control de las tareas realizadas, y sus duraciones.

Al final de cada evolución, se realizó un análisis de las métricas tomadas, con el fin de mejorar las estimaciones.

Finalmente se concluyó el proyecto con un análisis final, que fue entregado al cliente, y la documentación para la entrega en la facultad.

A lo largo del capítulo, se profundizarán los aspectos mencionados anteriormente.

7.2. Gestión de las comunicaciones

7.2.1. Comunicaciones dentro del equipo

A partir de abril, y durante el transcurso de todo el año, se definieron reuniones semanales con el tutor del proyecto. Dichas reuniones permitieron comunicar al tutor sobre las tareas realizadas y discutir las tareas a realizar para la siguiente semana. Esto fue de gran utilidad para mantener una disciplina de trabajo, ayudar a la organización del proyecto, y fomentar la realización de tareas y avance. Además, y de forma temprana, permitió poner sobre la mesa y discutir las distintas alternativas de acciones a realizar, y encaminar el proyecto, ayudando a prevenir y evitar errores que pudieran presentarse.

Dentro del equipo, se utilizó la aplicación móvil WhatsApp [31], que permitió tener una comunicación fluida, rápida y constante entre los integrantes. Se utilizó diariamente para comunicar decisiones, dudas o comentarios.

7.2.2. Partes interesadas del proyecto.

A continuación, se detallan los interesados más importantes para el proyecto, indicando su nombre dentro de la empresa (su rol) y su participación para la solución.

En primer lugar, los **tutores de la facultad pertenecientes a ORTs**, quienes ayudaron a formular y validar los distintos documentos y planes referentes al rol de cada uno de ellos.

Gerente de Sistemas y Est. Tecnológicos, fue el *Sponsor* por parte de la empresa para el proyecto. Quien decidió y validó las propuestas y alternativas definidas por el Gerente del Proyecto, así como los requerimientos, funcionalidades y soluciones propuestas para desarrollar.

Sub Gerente General, a quien se le fue presentando el avance del proyecto y validando las versiones del sistema móvil. También validó el documento inicial que detalla el funcionamiento general de la empresa, de tal forma que el equipo pudiera basarse en el mismo para la definición de soluciones.

Desarrolladores, Analista de la Base de Datos, y Administrador de Redes, quienes apoyaron en la definición y validación de la arquitectura propuesta, y comprensión de la lógica del sistema utilizado actualmente por la empresa (principalmente sobre cómo se conectan entre sí los distintos datos almacenados en la base de datos).

Gerente Comercial y de Servicios, Gerente de Recursos Humanos y Administración, Jefe del Departamento de Transporte, Jefe del Departamento de Producción, Jefe del Departamento de Operaciones, y Jefe del Departamento de Recaudación, fueron quienes ayudaron a conocer y entender cómo funciona C.U.T.C.S.A. en su ámbito general, sus procesos más importantes, y detección de problemas u oportunidades de mejora existentes.

A su vez, el **Jefe del Departamento de Producción** se encargó de la validación de las distintas versiones de la aplicación móvil a utilizar por los actores del sistema (Inspector, Largador, Conductor, Guarda y Conductor Cobrador).

La empresa **Handsoft** intervino en el proyecto para proporcionar dos servicios web esenciales para la solución.

Finalmente, los **inspectores, largadores, conductores, guardas, y conductores cobradores** ayudaron a priorizar requerimientos que les parecieron más útiles, así como probar las funcionalidades y usabilidad de la aplicación.

7.2.3. Resumen de comunicaciones

La siguiente tabla describe los elementos comunicados a los distintos interesados.

Elemento comunicado	Responsable a ser comunicado	Medio de comunicación	Frecuencia de comunicación.
Entrevistas iniciales para conocer el funcionamiento general de la empresa	A quienes se les realizó la entrevista	Reunión para las entrevistas, Correo electrónico para responder con las correcciones	Única vez al principio del proyecto

Descripción de la empresa a partir de entrevistas	Sub Gerente General	Reunión	Única vez luego de finalizar las correcciones de las entrevistas
Plan del proyecto	Gerente de Sistemas y Est. Tecnológicos	Reunión	Cada tres meses, con las actualizaciones realizadas
Planes de SCM, SQA y calidad	Gerente de Sistemas y Est. Tecnológicos	Reunión	Al principio del proyecto, una vez finalizados
ESRE	Gerente de Sistemas y Est. Tecnológicos, y Jefe del Departamento de Producción	Reunión/correo electrónico, respondiendo con las correcciones	Al inicio de cada evolución, una vez finalizado el documento
Documento de Arquitectura	Gerente de Sistemas y Est. Tecnológicos, y Desarrolladores de la empresa	Reunión	Al inicio de cada evolución, una vez finalizado el documento
Aplicación móvil	Gerente de Sistemas y Est. Tecnológicos, y Jefe del Departamento de Producción	Reunión	Al final del desarrollo en cada evolución
Aplicación móvil y resumen del proyecto	Sub Gerente General	Reunión	Al final del proyecto

Tabla 7-1 Comunicaciones con los distintos stakeholders del proyecto

7.3. Planificación de tareas y recursos

A continuación, se explican cada una de las actividades planificadas para el proyecto, agrupándolas tal como se muestra en la ilustración 7.2. En ella, se puede apreciar la planificación del proyecto, agrupadas en cinco categorías (Primera, Segunda y Tercera Evolución, Actividades académicas ORTs, y Actividades de apoyo), con las actividades más importantes.

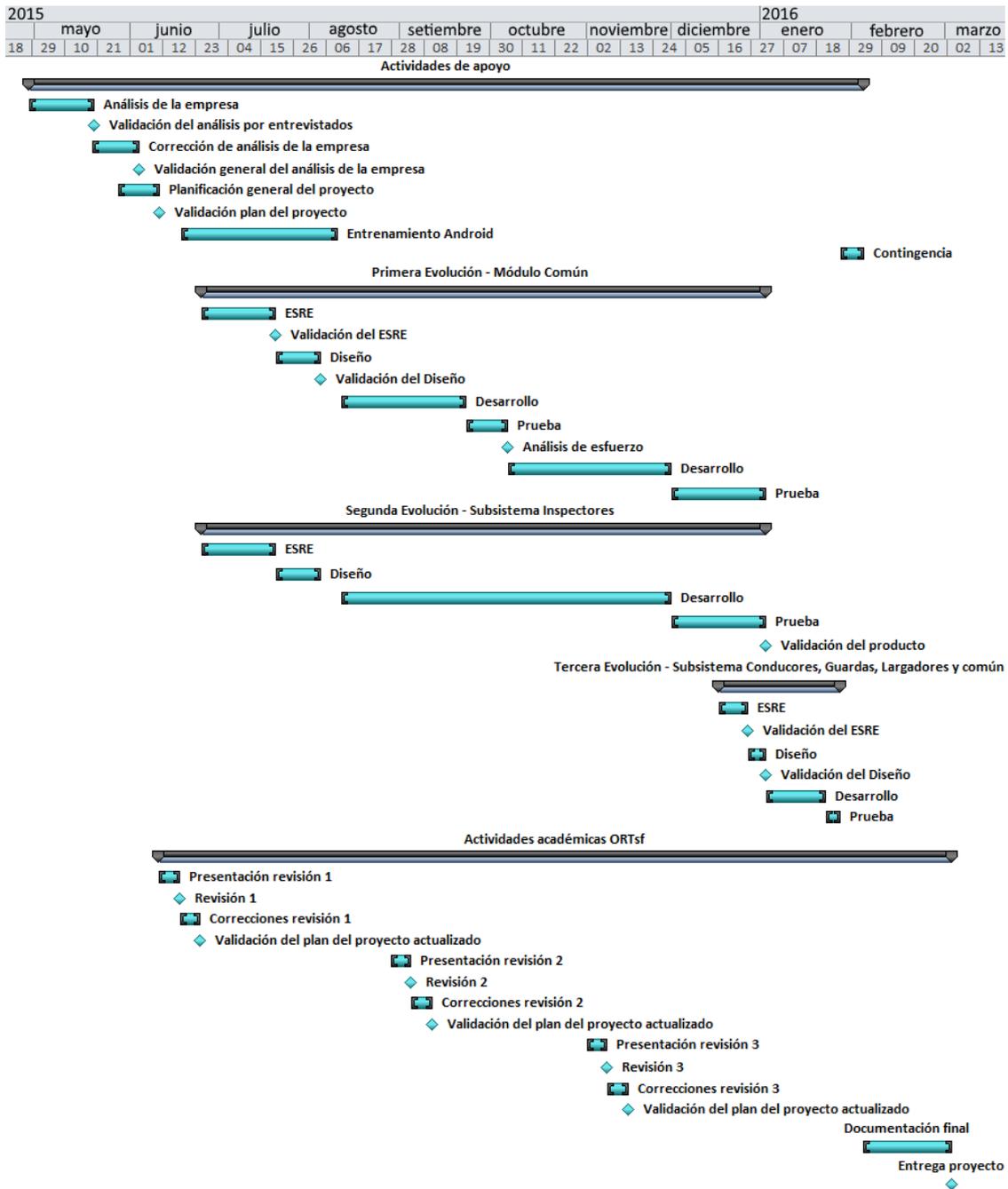


Ilustración 7-2 Diagrama de Gantt del cronograma del proyecto

Primera y Segunda Evolución.

Las primeras dos evoluciones del sistema a desarrollar (que hacen referencia al Módulo común y al módulo de Inspectores) se planificó para realizarse de forma concomitante. Se tomó esta decisión porque ya se tenía los requerimientos para el módulo de Inspectores previamente al comienzo del proyecto, pero era necesario realizar la base del sistema también (o sea, el módulo Común), ya que el módulo de inspectores se basaría en el mismo.

Al finalizar la especificación de requerimientos para ambas evoluciones, la cual se definió para las fechas entre fines de junio y comienzos de julio del 2015, se decidió que fuese entregado al cliente para ser validado y hacer las correcciones que entienda oportunas.

Luego se planificó continuar con la especificación del diseño durante el resto del mes de agosto, apoyándose en reuniones con personal de TI de la empresa. Se decidió así, finalizar la actividad con una entrega al cliente sobre el documento resultante para que sea validado.

A continuación, se determinó proceder con el desarrollo de ambas evoluciones, comenzando el 11 de agosto del 2015. En particular, se fijó que la primera evolución debería tener un pequeño corte en la mitad del desarrollo, el 21 de setiembre, para realizar las pruebas de los requerimientos desarrollados hasta el momento, culminando con una demostración al Jefe del Departamento de Producción para que valide lo realizado.

Esto permitió tomar métricas de la actividad y así mejorar la estimación del resto de los requerimientos a desarrollar y probar para ambos módulos de las evoluciones. De esta forma, se detalló finalizar el desarrollo el 29 de noviembre para comenzar las pruebas, cuya actividad se planeó para realizarse hasta el 31 de diciembre del 2015.

Para finalizar ambas evoluciones, se programó realizar una validación del desarrollo al Jefe del Departamento de Producción. En caso de necesitar realizar correcciones, de ser importantes, se realizarían en de forma simultánea con la tercera evolución, aumentando el esfuerzo para que no existan retrasos.

Tercera Evolución.

De forma concomitante al final de las evoluciones anteriores, a partir de mediados de diciembre, se planificó comenzar la segunda evolución del sistema, conteniendo el módulo de guardas, conductores y conductores cobradores, y el módulo de largadores, así como agregados de requerimientos para el módulo común. Dicha evolución se programó iniciar con el relevamiento de los requerimientos a desarrollar, por mediados de diciembre, culminando con la entrega del documento al cliente para su validación.

Se decidió realizar el desarrollo durante los primeros 20 días de enero, dedicando luego 5 días para realizar las pruebas. Hasta fines de diciembre no se tuvo una estimación, dado que se realizó luego de la especificación de requerimientos en dicha evolución. Su duración fue definida a partir de los requerimientos relevados, y el esfuerzo dedicado para las mismas actividades en las evoluciones anteriores.

De esta forma, la evolución se planificó finalizar el 25 de enero del 2016, para comenzar con la documentación a entregar en la facultad.

Actividades de apoyo.

Aquí se agruparon las actividades que brindaron ayuda para la realización del resto de las actividades del proyecto.

La planificación general del proyecto, incluyendo el aseguramiento de la calidad y la gestión de la configuración se planificó finalizar en mayo, una vez que se tenía una visión completa del problema. Al finalizar dicha planificación, se entregó al cliente para su evaluación y validación.

A raíz de que solo uno de los integrantes del equipo tenía conocimiento y experiencia en desarrollo para Android, y que la base del proyecto fue una aplicación móvil, se decidió generar un plan de entrenamiento para que el resto de los integrantes pudiera aprender a desarrollar. Dicho plan se definió realizarse luego de la generación de los planes más importantes, y su desarrollo se decidió realizarse en paralelo a las actividades de especificación de requerimientos y diseño provistos para el mes de julio, con el objetivo de culminar el entrenamiento para principios de agosto, a tiempo para el inicio de la actividad de desarrollo de la solución en la primera evolución.

Para el final del proyecto, se previó una contingencia de 8 días para el proyecto basados en los requerimientos relevados en la tercera evolución, y su correspondiente análisis de puntos de función y esfuerzo dedicado para las actividades en las evoluciones anteriores.

Actividades académicas ORTs.

Son las actividades que apuntan a entregas o actividades estipuladas por la facultad.

A lo largo del proyecto, se debieron realizar 3 revisiones y una entrega final. Las fechas exactas de las revisiones no se tenían hasta dos semanas antes de la realización de las mismas, a diferencia de la entrega final que desde el comienzo del proyecto se sabía que se sería el 3 de marzo del 2016.

Por cada revisión, se definió destinar una semana para la realización de la presentación, y luego una semana para realizar las correcciones indicadas por los revisores. La primera revisión se planificó para mediados de junio, la segunda para principios de setiembre, y la tercera para principios de noviembre. Se decidió que las tareas asociadas a las revisiones podrían adelantarse o retrasarse dependiendo de las fechas estipuladas por la cátedra para la realización de las mismas.

Al finalizar cada una de las correcciones de las revisiones, y en función del análisis realizado sobre el proyecto para las mismas, se decidió realizar actualizaciones al plan del proyecto con ajustes del cronograma, análisis de actividades, esfuerzo y seguimiento de riesgos. Dicho plan se definió entregarse al cliente para ser validado al final de cada revisión.

Finalmente, se planificó destinar el mes de febrero para la realización de la documentación necesaria para la entrega final del proyecto en la facultad.

Para ver el cronograma de forma más detallada, se puede consultar el [anexo del cronograma del proyecto](#).

7.4. Herramientas de seguimiento de tareas

El seguimiento de la ejecución del proyecto se realizó con un cronograma general de fases, evoluciones e hitos, utilizando la herramienta Microsoft Project 2010. El seguimiento de las tareas dentro de cada Evolución, se realizó utilizando la herramienta Manage It [32]. Esta combinación de herramientas permitió al equipo del proyecto contar con visibilidad general del proyecto y de los hitos previstos, y a su vez, hacer un seguimiento diario de las tareas de cada Evolución.

El cronograma realizado a través de Microsoft Project 2010 [33], permitió distinguir las fechas de inicio y fin tanto del planificado como del real para cada una de las tareas a realizar, así como el diagrama de Gantt creado por la herramienta para una visualización más gráfica del cronograma.

También se fueron registrando las tareas y horas insumidas en cada una de ellas por los integrantes del proyecto, permitiendo de esta forma tener un mejor seguimiento de las tareas pendientes, en proceso, y finalizadas. El registro de horas colaboró en la estimación de próximas tareas a realizar, permitiendo definir cuáles requerimientos realizar y cuáles no en función de su complejidad, teniendo en cuenta la priorización definida por el cliente para cada uno de ellos. La técnica utilizada fue Análisis de puntos de función, la cual se detalla más adelante dentro del capítulo.

Dicha organización de tareas a realizar comenzó siendo una planilla, denominada "ToDoList", creada en Google Drive [34], con una Hoja de cálculo de Google. La misma contenía el nombre de cada tarea, una lista de subtareas, su prioridad, cuándo se esperaba tener concluida la tarea, cuándo efectivamente se finalizó, el estado de la tarea (no iniciado, iniciado y finalizado), y el integrante del proyecto responsable por la tarea. Para un ejemplo de uso de la aplicación, ver el [Anexo 20 - ToDoList](#).

Una de las razones por la que se decidió realizar el registro de tareas en Google Drive fue por la posibilidad de tener múltiples usuarios visualizando y editando la planilla. Pero este tipo de formato era engorroso para crear una nueva tarea, definir sub tareas, y dificultaba el registro de horas por cada integrante. Esta fue la razón por la cual se decidió buscar una alternativa más funcional.

El registro de horas se realizaba también en Hojas de cálculo de Google, una planilla por cada integrante, con el objetivo de registrar por cada tarea cuántas horas insumió. Se debía registrar por cada día, las tareas realizadas, qué tipo de tarea era (gestión de proyecto, ingeniería de requerimientos, investigación y entrenamiento, arquitectura, desarrollo, prueba, SQA, SCM, revisiones y documentación, o retrabajo), y cuánto tiempo se dedicó. Pero solo uno de los integrantes hizo el registro, lo cual trajo problemas en la estimación y control de seguimiento del proyecto. Ir al [Anexo 19 – Registro de horas](#) para visualizar un ejemplo de uso.

Investigando en la web alguna herramienta que permitiera la gestión de tareas en un proyecto, se encontró Manage It, una aplicación web (que también puede descargarse para iOS y Android) muy completa que cumplía con las características que se buscaba. La misma permite registrar tareas y sub tareas de forma ágil en una interfaz bien organizada. A su vez, uno puede definir los responsables y fechas límites para cada una de forma tal que al usuario asignado se le notifique cuando la misma está próxima al límite. También permite crear notas (utilizada para dar una explicación más detallada sobre la tarea), adjuntar archivos asociados, y realizar comentarios. Todos los usuarios pueden acceder de forma concurrente a la aplicación y realizar modificaciones en tiempo real.

Como límite, la aplicación no provee una interfaz para el registro de horas, pero para solucionarlo, se utilizó la interfaz de comentarios para dichos registros: cada vez que se trabajaba sobre una tarea, al final del día se realizaba un registro con un comentario de la cantidad de horas dedicadas a la tarea. De esta forma, en el historial de comentarios de la tarea, se podía apreciar todos los registros de horas realizada por cada integrante. Este tipo de registro, y la exigencia por parte del gestor del proyecto para cumplirla, permitió tener datos de dedicación a ser utilizadas para la estimación y análisis. Para visualizar un ejemplo de uso, ir al [Anexo 18 – Aplicación Manage It](#).

Las tareas, asignaciones y fechas límite eran definidas por el gerente del proyecto y comunicado a través de dicha aplicación al resto de los integrantes. Las asignaciones se definieron en función del rol y habilidades del integrante del proyecto.

Las discusiones de tareas a continuar se realizaron en la reunión con el tutor. Un análisis más profundo de cada una, por parte del gerente del proyecto, permitía dividir la tarea en sub tareas y ser asignadas a cada integrante según su rol y fortaleza.

7.5. Estimaciones

Las técnicas de estimación utilizadas en el proyecto fueron: Juicio Experto, Analogía y *Bottom up*. Para calcular el tamaño del producto se utilizó la técnica de estimación temprana Puntos de Función y se realizó un primer desarrollo y prueba de algunas funcionalidades para poder tener el esfuerzo real y hacer analogía, y de esta forma estimar el esfuerzo requerido para el resto del proyecto.

Se tuvo en cuenta la fecha de entrega del proyecto, así como las revisiones que se debían realizar a lo largo del año.

A su vez, como todos los integrantes cursaron materias a lo largo del año, se debieron considerar las evaluaciones fijadas por la facultad para cada una de ellas. También se debía tener en cuenta las fechas especiales (vacaciones, feriados, fines de semana largos) que podían hacer que personal de la empresa no se encontraran disponibles.

Para definir la duración de las actividades, se tomó como suposición que cada integrante dedicaría al proyecto entre 15 y 20 horas semanales, tal como lo recomendaba la facultad.

En el [Anexo 5 en la sección Supuestos, restricciones y obligaciones](#) describe con más detalles las restricciones y supuestos tenidos en cuenta para la realización del plan del proyecto.

En conclusión, luego de los análisis realizados, se pudo determinar la realización de un total de 319 puntos de función para el proyecto, divididas en las tres evoluciones.

7.6. Seguimiento y evaluación de actividades

Existieron algunos desvíos con respecto a la planificación. A grandes rasgos, dicho desvío se puede apreciar en la ilustración 7.3, con fechas de inicio y fin aproximadas, describiéndose la planificación de color verde (barra superior), y la ejecución en color azul (barra inferior).

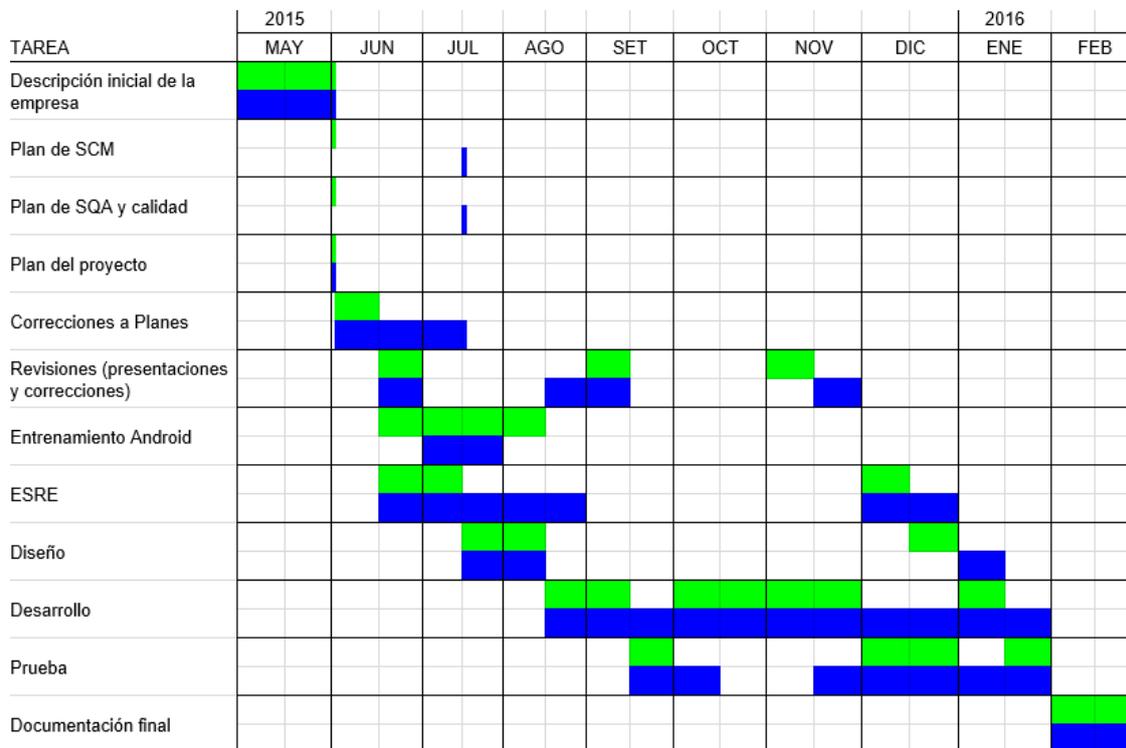


Ilustración 7-3 Comparación de alto nivel de la planificación y ejecución de actividades

El primer desvío importante que hubo fue con los planes de SCM, SQA y calidad, que se retrasaron por evaluaciones que se tuvieron por la facultad (entregas, parciales y exámenes), y del cual se decidió continuar con el resto de las actividades, tales como el entrenamiento, especificación de requerimientos y diseño

El siguiente fue el entrenamiento para aprendizaje en Android. Se comenzó más tarde a causa de la actividad de ingeniería de requerimientos que comenzó al mismo tiempo que lo planificado para dicho entrenamiento. Se ejecutó el plan durante un mes, pero a causa del resto de las actividades del proceso de ingeniería, principalmente del retraso en la especificación de requerimientos, debió suspenderse. En ese momento surgió la decisión de que quien tenía conocimientos sobre desarrollo en dicha plataforma, realizara todo el desarrollo de la aplicación móvil.

La especificación de requerimientos tuvo un retraso en las tres evoluciones a causa de evaluaciones en la facultad por los fines del semestre. Su retraso fue mayor del que se estimó, lo cual obligó a que dicha actividad se extendiera en el tiempo.

Para la primera y segunda evolución, no se retrasó el relevamiento, sino la especificación en sí, por lo tanto, se decidió continuar la actividad de diseño de forma concomitante.

Para la tercera evolución, la actividad de diseño sí dependió de la especificación de requerimientos, y por dicha razón también se retrasó.

La actividad de desarrollo tuvo un esfuerzo mayor al planificado, que se extendió sin detenimiento desde mediados de agosto hasta fines de enero. Dicho esfuerzo se debió mayoritariamente al retrabajo realizado para corregir los defectos detectados en la actividad de *testing*, o durante el desarrollo en sí. También afectó el desarrollo para satisfacer el requerimiento de seguridad, del cual requirió mayor tiempo del planificado. En la búsqueda de disminuir el retraso, se decidió aumentar la dedicación.

Finalmente, la actividad de *testing* necesitó mayor tiempo del planificado, principalmente para la primera evolución, ya que no se tenía un análisis previo para mejorar su estimación. Para las siguientes dos evoluciones, dicha actividad comenzó antes de lo planificado porque, a medida que se fue finalizando el desarrollo del *frontend*, se adelantó la actividad de *testing*.

Para ver el cronograma más detallado, ir al anexo [Cronograma del proyecto](#).

7.7. Gestión de riesgos

Para el seguimiento de los riesgos, se realizaron análisis, en algunas etapas del proyecto, con el fin de gestionarlos y de evaluar si los mismos se fueron mitigando o eliminando. Para esta sección, llamamos *etapas* a los siguientes hitos: la identificación y análisis de los riesgos al inicio del proyecto, y luego a cada revisión realizada en la facultad. En cada uno de ellos se monitorearon los riesgos planteados en este sub capítulo, en las cuales, a su vez, se registró dicho análisis en el plan del proyecto que se entregaría al cliente.

Para la identificación de riesgos, el equipo tuvo una reunión inicial para discutir los posibles riesgos pertenecientes al proyecto, y cuantificarlos para tener un seguimiento de los seis más importantes, es decir, con mayor magnitud. En el anexo [Análisis de otros riesgos](#) se puede apreciar la lista detallada del resto de los riesgos analizados al inicio del proyecto.

Sobre el análisis cuantitativo, para cada riesgo identificado al inicio del proyecto, se determinó el impacto y probabilidad de que los mismos sucedieran, basándose en las experiencias de los integrantes del equipo. A continuación, se calculó la magnitud de cada uno, en función de la fórmula:

$$\text{Impacto} * \text{Probabilidad de ocurrencia} = \text{Magnitud}$$

Ecuación 7-1 Calculo de Magnitud de un riesgo

En cada análisis, tal como se aprecia en la gráfica 7.1, la magnitud responde al análisis del riesgo en cada etapa. En caso de haberse materializado el riesgo, no se verá reflejado en dicha tabla, sino que se describirá el caso más adelante. La misma gráfica describe, frente a cada riesgo, y en función a las características que tuvo el proyecto al momento de ser analizado, cuál fue su probabilidad de ocurrencia e impacto.

A continuación, se describe los principales riesgos identificados en el proyecto, y cómo fueron evolucionando.

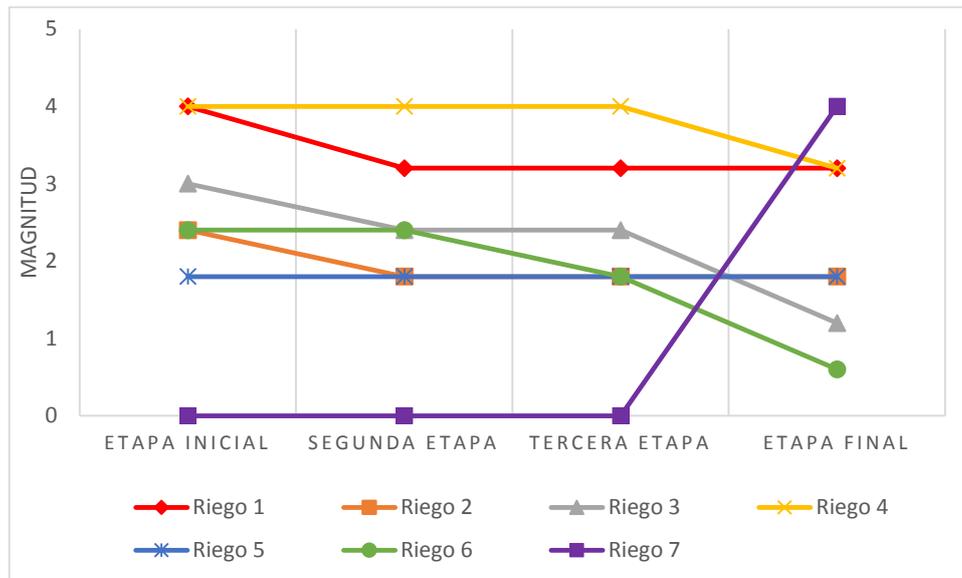


Gráfico 7-1 Seguimiento de la magnitud de los riesgos más importantes

Id. Riesgo	Descripción
Riesgo 1	Falta de conocimiento sobre desarrollo para Android de dos de los integrantes del proyecto.
Riesgo 2	Falta de conocimiento del contexto y requerimientos iniciales para el inicio del desarrollo del sistema.
Riesgo 3	Proyecto pensado para todas las áreas de la empresa, que sin un buen diseño y arquitectura puede generar retrabajo.
Riesgo 4	Falta de conocimiento sobre seguridad para sistemas informáticos.
Riesgo 5	Falta de disponibilidad de las partes interesadas del proyecto para reunirse, validar documentos, etc..
Riesgo 6	Fallas en la estimación del cronograma.
Riesgo 7	Dependencia sobre servicios o sistemas desarrollados por la empresa, o de empresas tercerizadas por la misma.

Tabla 7-2 Descripción de los riesgos más importantes

En la gráfica 7.1 se puede apreciar la evolución de cada riesgo a lo largo del proyecto en base a su magnitud (resultado del análisis del impacto y probabilidad de ocurrencia). Las referencias a los identificadores de los riesgos, se pueden apreciar en la tabla 7.3.

En primer lugar, se detectó el riesgo de falta de conocimiento en desarrollo para dispositivos móviles (riesgo 1), el cual desde el inicio del proyecto tenía un gran impacto al punto de retrasar el proyecto de forma significativa. Y era seguro que sucedería porque dos de los integrantes del proyecto jamás habían aprendido a desarrollar para Android, una de las restricciones del proyecto. Es por dicha razón que se decidió crear un plan de entrenamiento, el cual permitió disminuir la probabilidad de ocurrencia.

Pero a raíz de los retrasos en las actividades del proceso de ingeniería, se debió cancelar poco después de haber comenzado la actividad. De todas formas, se resolvió que el integrante con conocimiento en el desarrollo en Android fuera el responsable de la aplicación móvil, y el resto de los integrantes dedicaran su esfuerzo de desarrollo al *backend*. Esta decisión terminó siendo correcta dado que nunca sucedió que se necesitara asignar más recursos para el desarrollo de la aplicación móvil.

El riesgo referente a la falta de conocimiento del contexto y requerimientos iniciales (riesgo 2), desde el inicio, fue considerado que podría retrasar un poco el proyecto, siendo de una probabilidad muy alta. Es por dicha razón que se decidió comenzar con un análisis de la empresa. Se entrevistó a varios referentes de áreas y departamentos más significativos de la empresa con la intención de entender mejor el contexto, y relevar algunos posibles requerimientos. De esta forma, se tuvo un camino marcado con las posibilidades de desarrollo y ayuda que se podría ofrecer a la empresa, y así disminuir su probabilidad de ocurrencia. Manteniendo una fuerte comunicación con los distintos interesados para el proyecto, se logró mitigar el riesgo significativamente, pero siempre con la incertidumbre de cualquier malentendido o cambio que requiera el cliente.

Dicho riesgo se materializó al comenzar la tercera evolución, ya que el equipo malinterpretó el término “viaje” usado por la empresa, y teniendo que realizar las modificaciones y *refactoring* pertinentes. Dicho desvío fue de un total de 5 horas de esfuerzo. De todas formas, su reestructuración no retrasó significativamente el resto de las actividades.

El riesgo de inestabilidad de la arquitectura y diseño (riesgo 3) estuvo al margen a lo largo de todo el proyecto, siendo su impacto crítico con una mucha probabilidad de que suceda. Con la fuerte comunicación con el cliente, y entregas constantes para validación, se logró ir disminuyendo su probabilidad de que este riesgo se materialice. Recién cuando se finalizó la tercera evolución, se pudo concluir que la arquitectura del sistema, analizada al principio del proyecto, fue la correcta, ya que no necesitó ser modificada a diferencia de lo indicado en el riesgo 2.

La falta de conocimiento sobre seguridad (riesgo 4) fue uno de los riesgos de mayor magnitud, el cual se mantuvo de esa forma a lo largo de todo el proyecto. Su impacto era crítico, ya que podría detener el proyecto dado que era una restricción para la solución, y era altamente probable que ocurriera, a pesar de los conocimientos básicos que el equipo tenía gracias a materias dictadas en la facultad. Se logró mitigar ligeramente el riesgo a raíz de que uno de los integrantes se enfocó en la solución de los problemas de seguridad en la etapa de desarrollo, con ayuda de expertos y profesores de la Universidad ORT. Y fue lo que efectivamente sucedió: la actividad de desarrollo se retrasó a raíz de este problema, y la única solución fue aumentar el esfuerzo para las tareas. Adicionalmente, se contó con el apoyo del Administrador de Redes de la empresa, que colaboró en la mitigación del riesgo.

La falta de disponibilidad de las partes interesadas para el proyecto (riesgo 5) fue un riesgo que se mantuvo constante a lo largo de todo el proyecto, por la incertidumbre de si cada interesado estaría disponible para el momento en que se necesitaba. Y, a pesar de que uno de los integrantes del proyecto trabaja dentro de la empresa, en muchos casos los interesados podían estar de licencia o trabajando en temas críticos para la empresa, y no podrían reunirse. Es por dicha razón que era muy probable que dicho riesgo ocurriera, con un impacto que podría retrasar el proyecto.

Dado que en muchos casos las licencias podrían ser variables (comenzar en distinta fecha a la estipulada) y podrían suceder muchos imprevistos por lo cual las reuniones debieran cancelarse (principalmente con personas de cargos importantes en la empresa), se buscó agendar reuniones lo más temprano posible, de forma anticipada a la actividad que dependió de ellas, y así no retrasar la misma.

El riesgo por falla en la estimación del cronograma comenzó siendo de alta probabilidad de ocurrencia e impacto importante, pudiendo retrasar el proyecto. Para ir disminuyendo la probabilidad de ocurrencia, se realizaron los análisis de puntos de función y registro de esfuerzo, con el fin de tener elementos más sólidos para la estimación. De esta forma, con el primer análisis se redujo un poco la probabilidad de ocurrencia en la tercera etapa con la primer y segunda evolución, y más aún en la tercera etapa, dentro de la tercera evolución.

Finalmente, se debió sumar el riesgo de dependencia de sistemas y desarrollos externos al proyecto, que surgió al final de la tercera etapa, mientras se desarrollaban algunos de los requerimientos de la segunda evolución (referentes a la evolución de inspectores). Este riesgo no se había tenido en cuenta al principio del proyecto, el cual se consideró que podría detener el proyecto, y con una alta probabilidad de ocurrencia (porque a pesar de que la empresa que estaba desarrollando los servicios necesarios para nuestro proyecto aseguró tenerlos terminados antes de la finalización del proyecto, no teníamos la certeza). Para el comienzo de la etapa final, ya cerca de la finalización del desarrollo de la segunda evolución, recibimos los servicios que eran necesarios para terminar con dicho desarrollo.

Para ver con más detalle los planes de respuesta y contingencia usados para el resto de los riesgos analizados, ver el anexo [Análisis de otros riesgos](#).

7.8. Métricas

7.8.1. Métricas generales

Desvío entre el esfuerzo planificado y el realizado.

Desde comienzos del proyecto se acordó, entre todos los integrantes del proyecto, dedicar entre 15 y 20 horas semanales, tal como lo sugería la facultad. El registro de horas dedicadas para cada una de las tareas comenzó el 6 de abril del 2015, y se realizó el seguimiento de tareas hasta fines de enero del 2016.

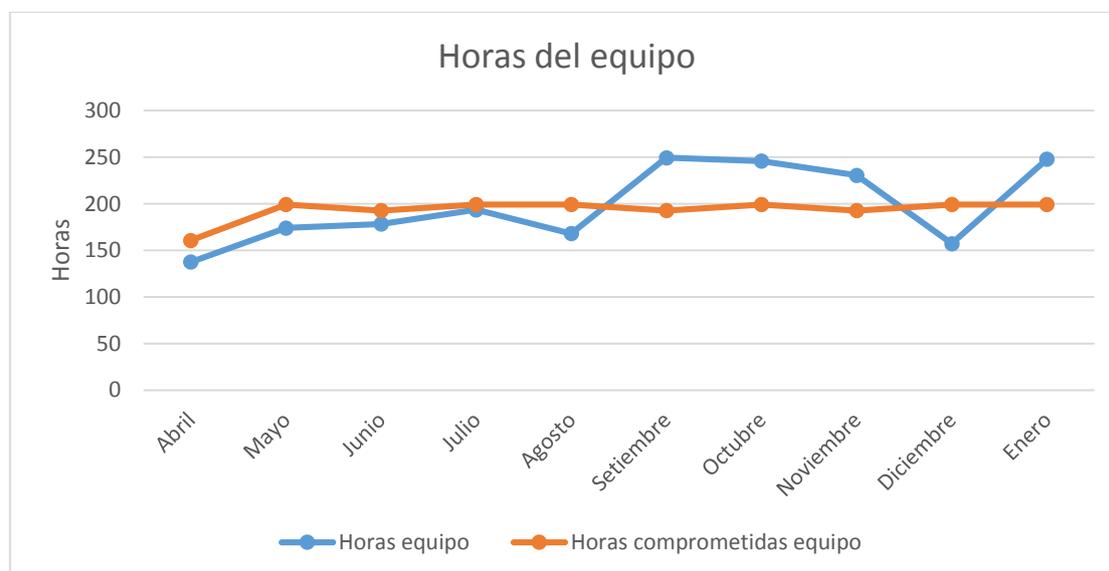


Gráfico 7-2 Desvío entre el esfuerzo planificado y realizado por el equipo

En la gráfica 7.2 se puede observar el esfuerzo mínimo comprometido por el equipo (en anaranjado) y el que realmente realizó en cada mes (azul). En los meses de abril, mayo y junio, dos de los integrantes del proyecto aún no registraban sus horas, por lo que se realizó una extrapolación en función al registro de los meses de julio y agosto, meses cuya dedicación fue similar al de los meses anteriores.

Los primeros meses, donde se realizaron muchas reuniones con personal de la empresa, no se logró cumplir con la dedicación comprometida. Se realizaban una o dos reuniones por semana (por problemas de disponibilidad de los entrevistados), y, a su vez, por las distintas evaluaciones que cada uno de los integrantes tenía en la facultad, no se pudo aumentar la dedicación.

De todas formas, en los momentos en que sí se pudo dedicar tiempo para el proyecto, se aprovechó para crear prototipos de distintas tecnologías a usar dentro de Android, de forma de anticipar posibles soluciones para el desarrollo.

Comparado con la dedicación que se estaba teniendo hasta el momento, existió una pequeña caída en agosto, a raíz de la etapa de análisis del diseño de la solución.

Tal como se puede apreciar, la dedicación aumentó significativamente entre los meses de agosto y setiembre, donde se comenzó con la actividad de desarrollo. Se decidió aumentar el esfuerzo para recuperar las horas de los meses en que no se cumplió con el esfuerzo comprometido.

Luego existió un valle en el mes de diciembre, a raíz de los parciales que se debieron realizar para la facultad, fiestas y fechas especiales, en las que se perdió dedicación para el proyecto.

Finalmente, se repuntó para finalizar a tiempo con los requerimientos propuestos por el cliente, tanto en el desarrollo como las pruebas.

De forma general, la siguiente tabla detalla los principales datos del proyecto.

Concepto	Valor
Esfuerzo total comprometido	1935 horas
Esfuerzo total realizado	1983 horas
Promedio de esfuerzo semanal grupal	46 horas
Promedio de esfuerzo semanal individual	15,4 horas

Tabla 7-3 Información general

De esta forma, podemos concluir que el equipo logró permanecer con un esfuerzo promedio dentro del rango sugerido por la universidad, superando así el esfuerzo comprometido para el proyecto.

Esfuerzo por área.

A lo largo del proyecto, las horas de dedicación fueron registrados en fechas pertenecientes a las distintas áreas de ingeniería y apoyo. La distribución de dicha dedicación se puede visualizar en la gráfica 7.3.

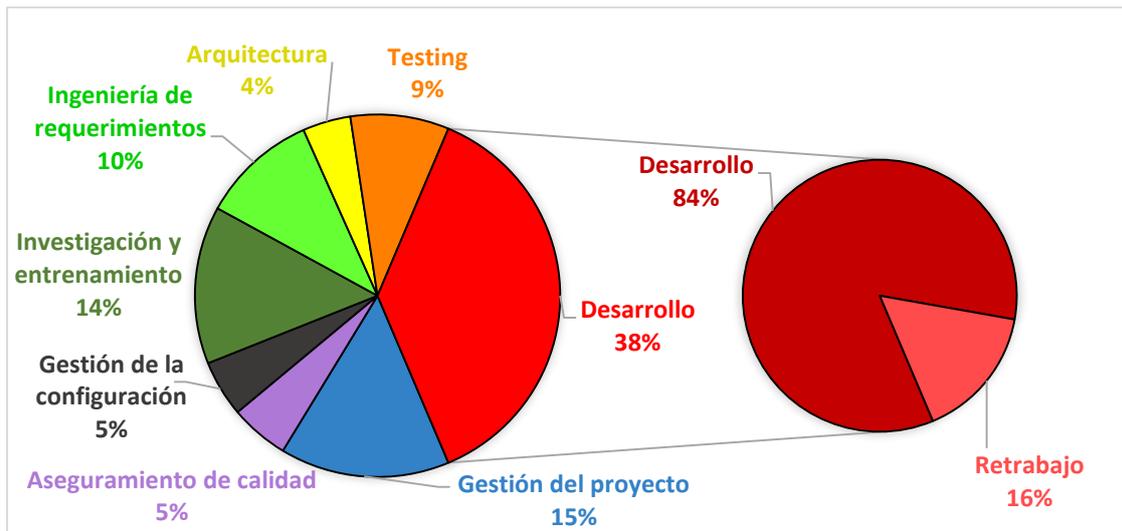


Gráfico 7-3 Comparación de dedicación de las actividades principales

Las actividades del área de ingeniería, que incluyen Ingeniería de requerimientos, Arquitectura, Desarrollo y *Testing*, fueron las que más dedicación tuvieron: 60,5%. Por otro lado, las actividades del área de apoyo, que incluyen Gestión del proyecto, Aseguramiento de calidad y Gestión de la configuración, tuvo una dedicación del 25,4% con respecto a la otra área. En particular, se consideró las actividades referentes a la Investigación y entrenamiento, que tuvo un esfuerzo importante en el proyecto con un 14,1%.

La actividad de Desarrollo fue la que tuvo mayor dedicación, 38%. Desglosando un poco dicha actividad, se puede apreciar que, de dicho desarrollo, el 84% correspondió al desarrollo en sí, mientras que el restante 16% fue retrabajo por correcciones.

Luego, la gestión del proyecto fue la segunda actividad que más esfuerzo tuvo, de un 15%, a raíz de todos los análisis realizados para el seguimiento de actividades, pronóstico del esfuerzo que llevarían cada una de las tareas a realizar, y reuniones tenidas semanalmente con el tutor (las cuales solían durar aproximadamente una hora).

Con respecto a las horas de investigación y entrenamiento, 83% corresponde a la investigación, mientras que el 17% restante al entrenamiento en Android (siendo este último un 2% del total invertido en el proyecto). A pesar de tomar la decisión de cancelar el entrenamiento, siendo así un tiempo perdido, el total invertido fue mínimo comparado con las horas dedicadas al resto de las actividades del proyecto.

Desvío entre el esfuerzo planificado y el realizado, en las actividades del proceso de ingeniería, por el equipo en cada evolución.

A lo largo de todas las actividades del proyecto, tal como se aprecia en la gráfica 7.4, las horas dedicadas a cada actividad fue menor a la planificada. Esto se dio de esta manera porque, a medida que los integrantes del equipo trabajaron en el proyecto, fueron obteniendo experiencia en las distintas actividades, y así aumentando su velocidad de producción.

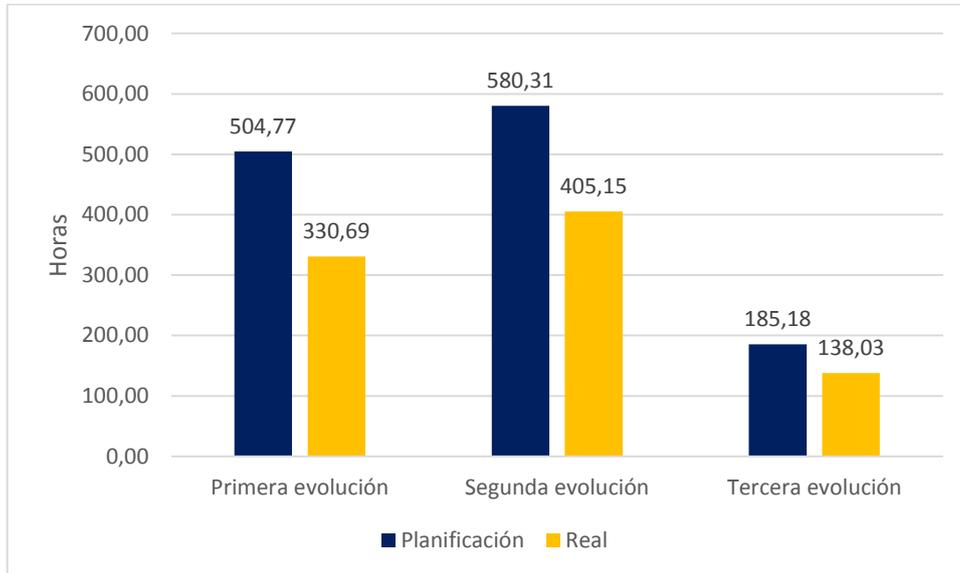


Gráfico 7-4 Desvío entre el esfuerzo planificado y real por evolución

De esta forma, tal como se describe en la gráfica 7.5, el proyecto fue disminuyendo su desvío con respecto al esfuerzo estimado, concluyendo que la estimación fue mejorando evolución tras evolución.

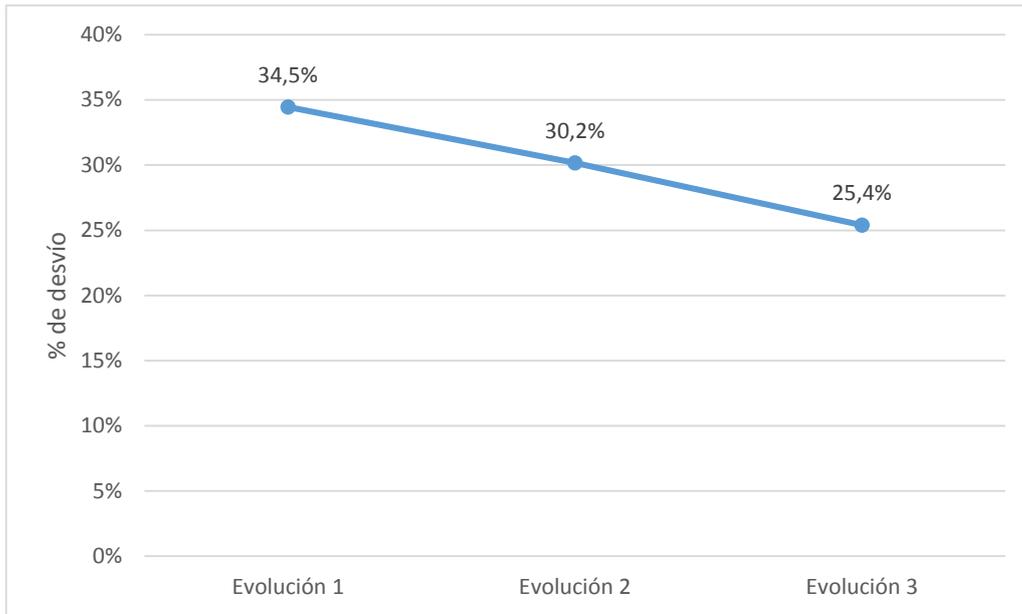


Gráfico 7-5 Comparación entre la planificación y la dedicación real para cada evolución, de las actividades del proceso de ingeniería

A continuación, se describen para cada actividad del proceso de ingeniería, una comparación entre las horas planificadas y las reales en cada evolución.

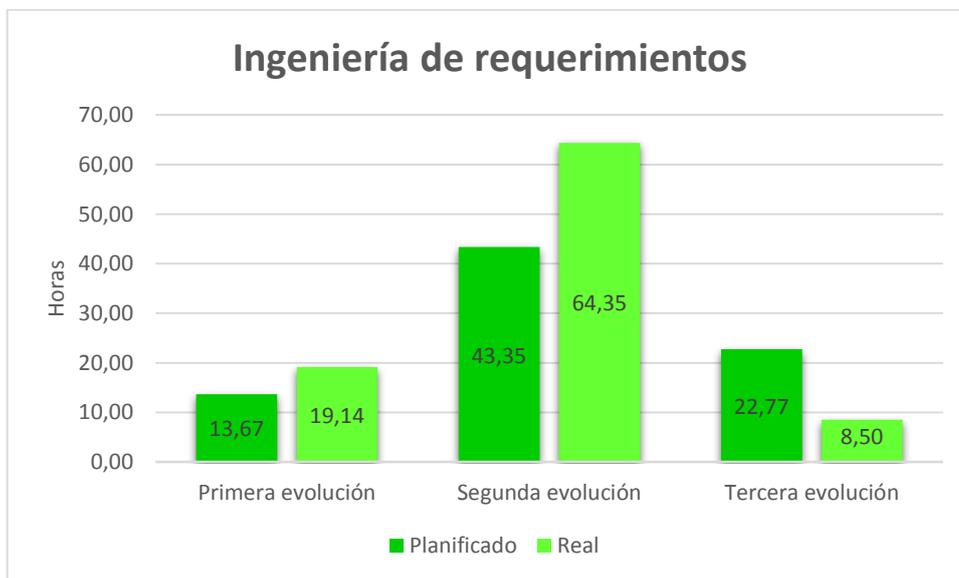


Gráfico 7-6 Comparación entre horas planificadas y reales para la Ingeniería de requerimientos en cada evolución

En la gráfica anterior se puede apreciar que, para la segunda y tercera evolución, existió un desvío negativo y positivo respectivamente. El primero, el cual sucedió durante la segunda evolución, fue a raíz de la cantidad de reuniones que se tuvieron con el Jefe del Departamento de Producción para que fuera validando, corrigiendo y agregando nuevos detalles al desarrollo de los requerimientos.

Por otro lado, en la tercera evolución, el desvío fue por una dedicación menor a la planificada, pero el mismo no afectó al resultado final. Esto sucedió por una falla en la estimación, la cual no fueron necesarias tantas horas para la actividad.

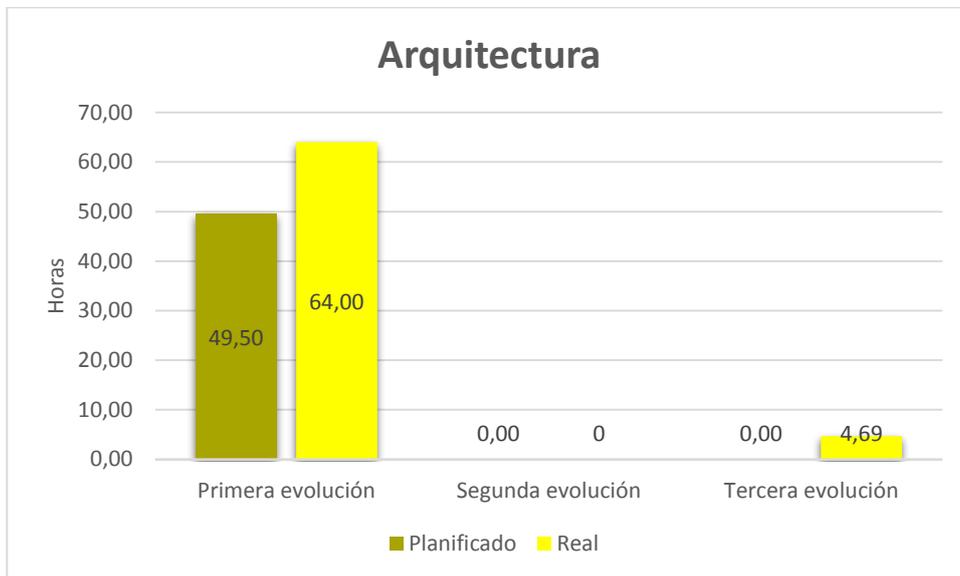


Gráfico 7-7 Comparación entre horas planificadas y reales para la arquitectura en cada evolución

Para el caso de la arquitectura, existió un pequeño desvío en la primera evolución a raíz de la falta de métricas para la estimación.

Por otro lado, en la tercera evolución, no se esperaba realizar ninguna modificación al diseño de la solución. Pero un error de concepto sobre los viajes y servicios (terminología interna de la empresa), obligó a realizar las correcciones pertinentes al documento y desarrollo.

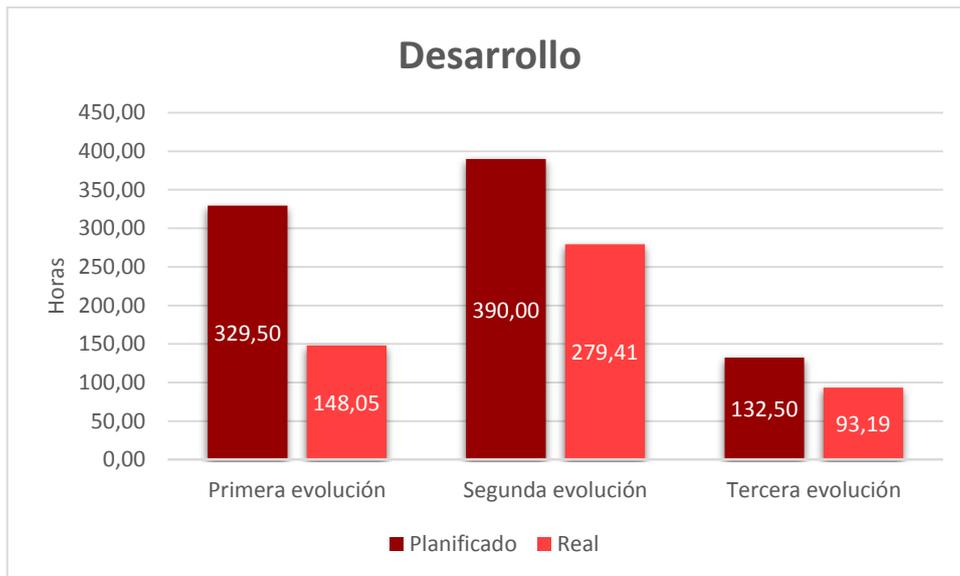


Gráfico 7-8 Comparación entre horas planificadas y reales para el Desarrollo en cada evolución

La estimación del desarrollo fue mejorando evolución tras evolución. La primera evolución se realizó sin ninguna métrica de estimación, por lo que el desvío fue mucho más significativo. A partir del análisis de puntos de función y el registro de horas, permitió ir mejorando poco a poco dicha estimación. Como se puede apreciar, el desvío entre la planificación y lo que realmente se hizo para cada evolución fue del 55%, 28% y 30% respectivamente, lo cual permite concluir que el análisis realizado mejoró significativamente la estimación.

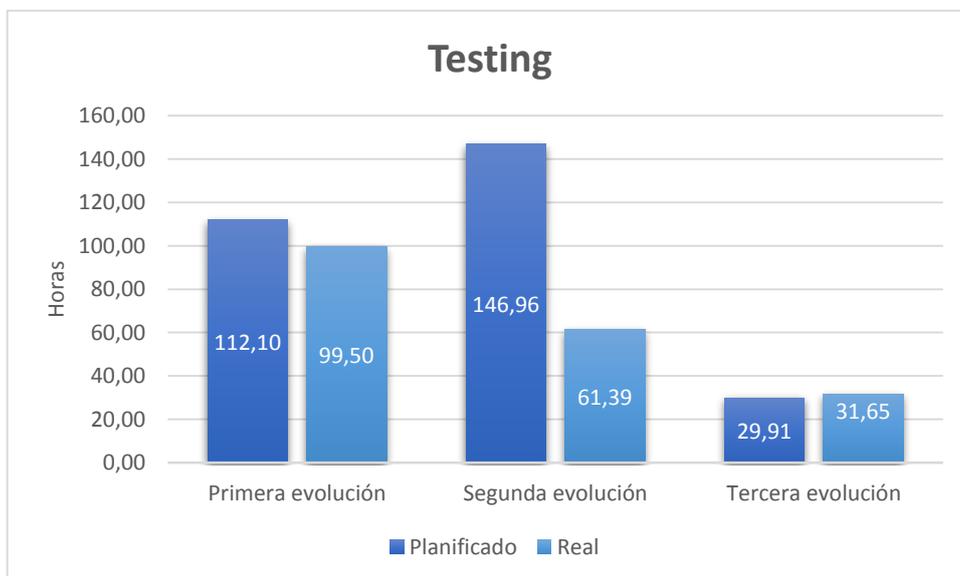


Gráfico 7-9 Comparación entre horas planificadas y reales para el Testing en cada evolución

Finalmente, el *testing* sufrió un desvío importante en la segunda evolución, que pudo haber sido producto de la cantidad de requerimientos desarrollados en dicha evolución comparado con la primera (más del doble, en puntos de función). También pudo haber

sido por el aumento de experiencia por parte del equipo para realizar las pruebas, que permitió finalizarlas con un menor esfuerzo del previsto.

Para la tercera evolución, analizando el esfuerzo realizado en la evolución anterior, se pudo mejorar la precisión de la estimación.

7.8.2. Métricas de la etapa de desarrollo

Puntos de función por cada módulo.

Como se describió anteriormente, para mejorar la estimación para la dedicación de los distintos requerimientos a desarrollar para la solución, se realizó un análisis de Puntos de Función combinado con el registro de horas realizado por cada integrante del proyecto para las distintas actividades de ingeniería (especificación de requerimientos, diseño, desarrollo y pruebas). En la tabla 7.5 se puede apreciar la cantidad de puntos de función analizados para cada requerimiento especificado para la solución.

Módulos	Requerimiento	Puntos de Función		
Común	G-RF1: <i>Log In</i>	8	92	
	G-RF2: <i>Log Out</i>	3		
	G-RF3: Mostrar Menú	11		
	G-RF4: Reportar un error en la aplicación	13		
	G-RF5: Recepción de notificaciones	8		
	G-RF6: Gestión de notificaciones	21		
	G-RF7: Visualización de noticias	8		
	Ver mapa	20		
Inspectores	C2-RF1: Ver información de paradas	41	195	319
	C2-RF2: Visualizar el recorrido de cierta línea en un mapa	12		
	C2-RF3: Mostrar arribo de líneas a la parada	15		
	C2-RF4: Ubicar coche en un mapa	37		
	C2-RF5: Ver ruta entre un origen y un destino	22		
	C2-RF6: Ver información de puntos de interés	17		
	C2-RF7: Control del servicio	42		
	C2-RF8: Mostrar horario planificado de una línea	9		

Conductores, guardas y conductores cobradores	C3-RF1: Consultar servicios asignados	17	17	
Largadores	C4-RF1: Modificar servicios	15	15	
	C4-RF2: Alta servicio			
	C4-RF3: Baja servicio			

Tabla 7-4 Puntos de función de cada uno de los requerimientos especificados

De esta forma, se planificó desarrollar 319 puntos de función, según el análisis realizado por el gerente del proyecto, con el fin de tener un conjunto de requerimientos completos para que los distintos funcionarios pudieran utilizar en el sistema.

Para ver el análisis de Puntos de Función de forma más detallada, consultar [Anexo 22 – Análisis de puntos de función](#).

Acumulación de Puntos de Función realizados en cada evolución.

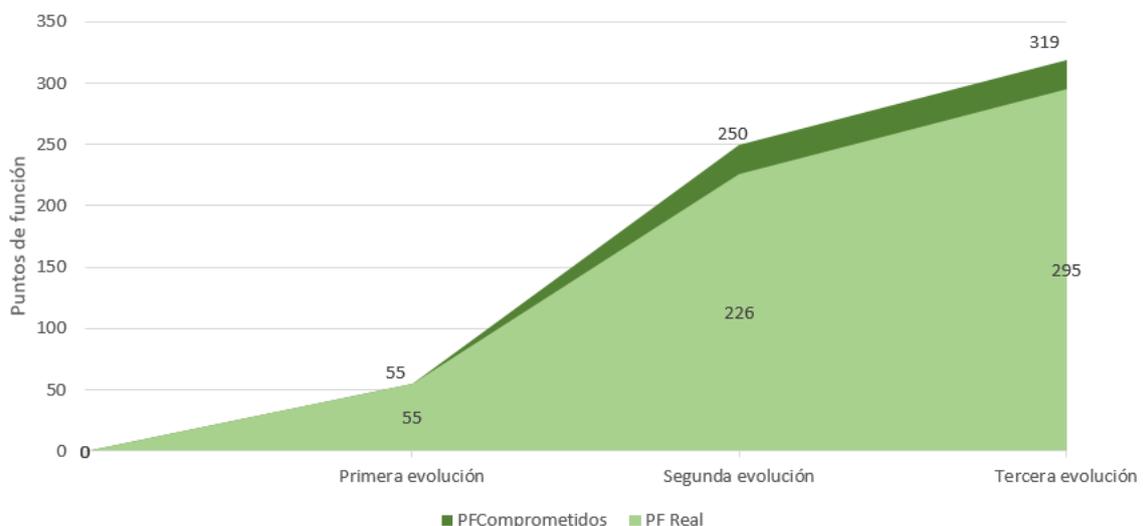


Gráfico 7-10 Acumulación de Puntos de función por evolución

Para la primera y tercera evolución, se logró cumplir con el desarrollo de todos los requerimientos definidos. Pero existió un desfase en la segunda evolución ya que un requerimiento no realizó, y otro no se logró completar, ambos correspondientes a las funcionalidades para los inspectores. El primero fue el *C2-RF5: Ver ruta entre un origen y un destino*, funcionalidad poco importante para los usuarios, que se decidió cancelar su desarrollo para dar lugar a un comienzo más temprano para la tercera evolución. Por otro lado, la funcionalidad *C2-RF7: Control del servicio* no logró completarse por

restricciones externas al proyecto. Dicha funcionalidad depende de una tarjeta puente para transportar los datos que contiene la máquina expendedora hacia el dispositivo. El protocolo definido por la Intendencia de Montevideo, especifica los datos a cargar en la tarjeta, pero muchos datos necesarios para el control e inspección a bordo de los ómnibus no están contemplados en el mismo, por lo que no se pudo completar el desarrollo hasta que se encuentre una solución por parte de C.U.T.C.S.A. y la Intendencia.

Productividad (PF/H) realizado en cada evolución.

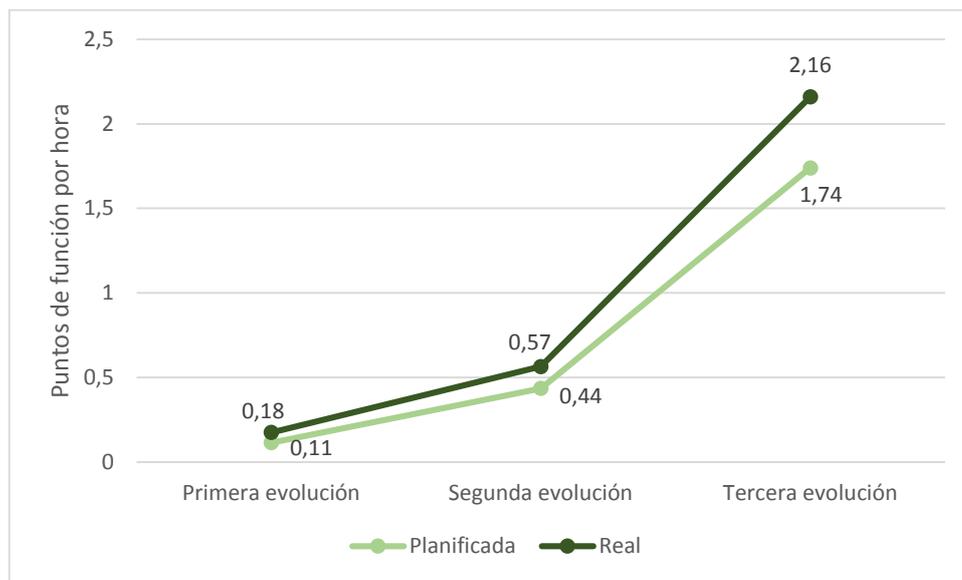


Gráfico 7-11 Productividad de las actividades del proceso de ingeniería

Tal como se aprecia en la gráfica 7.11, a medida que se realizaron las evoluciones, la velocidad del equipo para realizar las actividades del proceso de ingeniería fue mejorando lentamente con respecto a la planificación, pero significativamente con respecto a las etapas anteriores. La velocidad que realmente tuvo el equipo en la última evolución, con respecto a la primera, fue 12 veces mayor. Esto sucedió porque, a medida que se avanzó en el proyecto y se obtenía mayor experiencia en las actividades, la velocidad aumentó significativamente, pudiendo realizar una mayor cantidad de requerimientos en menor tiempo.

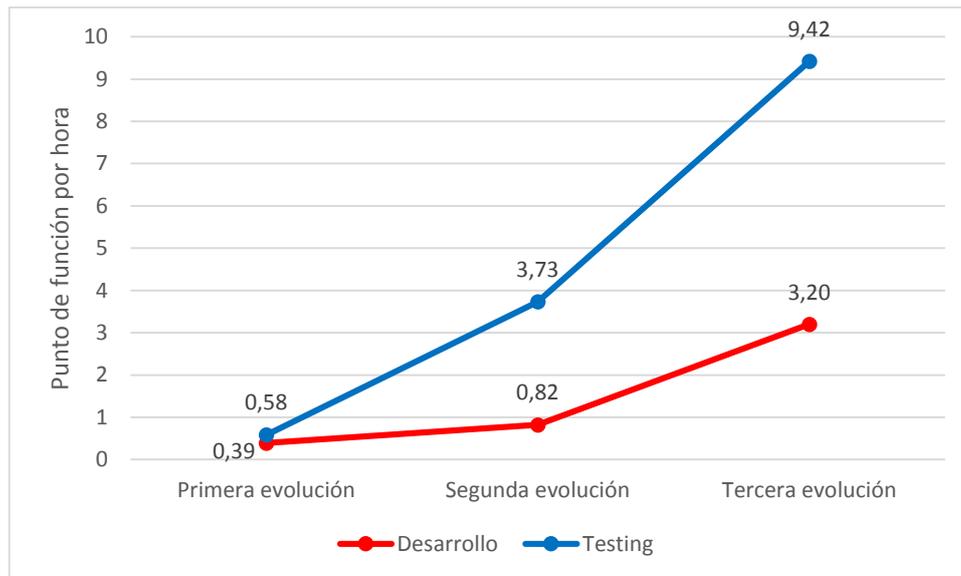


Gráfico 7-12 Productividad real de las actividades de desarrollo y testing

En particular, en la gráfica 7.12 se puede apreciar la velocidad de desarrollo y *testing* para los requerimientos en cada evolución. En ambos casos, fueron aumentando exponencialmente, con una curva menos pronunciada para el caso del desarrollo por el retrabajo que se debió realizar.

En definitiva, y comparando la última y primera evolución, el desarrollo aumentó su velocidad 8 veces, mientras que el *testing* aumentó 16 veces.

7.9. Principales decisiones de gestión

A continuación, se resumen las decisiones más importantes tomadas por el equipo con respecto a la gestión.

Reuniones semanales con tutor: tener reuniones periódicas con el tutor del proyecto, permitió que el mismo supiera en todo momento las actividades que el equipo estaba realizando, y analizar las próximas decisiones a tomar.

Aumentar el esfuerzo por incumplimiento de dedicación: luego de detectar que en los primeros meses el equipo no estaba cumpliendo con el esfuerzo comprometido, se decidió aumentarlo para los siguientes meses, y mantenerlo de esta manera hasta el final del proyecto.

Dos evoluciones al mismo tiempo: tal como se explicó a lo largo del capítulo, existió una restricción técnica de comenzar con el desarrollo de la primera evolución, y una necesidad de comenzar con el desarrollo de la aplicación móvil.

Análisis de puntos de función: junto con el análisis de métricas tomadas durante el proyecto, permitió mejorar satisfactoriamente la precisión del esfuerzo a dedicar para las distintas actividades del proceso de ingeniería.

Entrenamiento Android: a pesar de que con esta actividad se pretendía eliminar el riesgo de que el desarrollo de la aplicación móvil se retrasara, terminó siendo una decisión equivocada. Designar un solo integrante como recurso para el desarrollo de la aplicación fue suficiente, y el mismo jamás se retrasó con respecto al desarrollo del *backend*.

Realización de prototipos de tecnologías móviles: fue una muy buena decisión para anticipar y evitar retrasos por desconocimiento tecnológico para las distintas soluciones, tanto móviles como para el *backend*.

7.10. Conclusiones y lecciones aprendidas

Con respecto a la gestión de proyectos, existieron muchos desafíos que permitieron ofrecer experiencias a los integrantes del proyecto. A continuación, se resumen las actividades más relevantes.

Gestión de riesgos: un análisis previo sobre los posibles riesgos, sumado a un constante análisis y seguimiento, permitió anticipar posibles problemas que pudieron afectar el avance del proyecto, y en muchos casos, mitigarlos.

Gestión del esfuerzo: el uso de planillas online para la distribución de tareas y registro de horas dentro del proyecto dio más problemas que soluciones, y la búsqueda de alternativas fue esencial para mejorar la eficiencia y eficacia dentro de este aspecto.

Evaluación de las evoluciones: la combinación del análisis de puntos de función con el registro de horas, fue muy importante para realizar mejoras a las estimaciones y, de esta forma, realizar mejoras al proceso de desarrollo.

Gestión del alcance: a través del análisis descrito anteriormente, se pudo mejorar, evolución tras evolución, el alcance que el equipo podría lograr en base al tiempo que se tenía para el desarrollo de la solución, llegando a desarrollar un total de 295 puntos de función.

Valores generales de la gestión del proyecto: en particular, se desea destacar algunas métricas tomadas a lo largo del proyecto.

1. Esfuerzo promedio por persona por evolución: 324 horas.

2. Velocidad de desarrollo promedio: 1,41 PF/hora
3. Esfuerzo total dedicado al proyecto: 1983 horas.

8. Aseguramiento de calidad

En este capítulo se presentan los conceptos de calidad definidos para realizar el aseguramiento de la calidad del proyecto. El mismo está compuesto por los objetivos de la calidad seguido por las 3 características principales en la cual se basó la calidad del proyecto: las actividades y tareas del aseguramiento de calidad, el *testing* y la usabilidad. Para finalizar se muestran métricas obtenidas.

8.1. Objetivos de la calidad

El propósito del área de aseguramiento de la calidad es definir actividades y tareas necesarias, y generar un marco de trabajo que sirva de guía para el desarrollo de los diferentes puntos del proyecto. De esta manera, se logra cumplir con los objetivos del proyecto, así como los objetivos académicos establecidos.

Con respecto al producto, el objetivo es validar los requerimientos a través de diferentes técnicas.

8.2. Aseguramiento de la calidad

Al comienzo del proyecto, el equipo definió un plan de aseguramiento de calidad del software (Plan de SQA), donde se establecen un conjunto de pautas para guiar las actividades de aseguramiento de calidad a cargo del Líder de SQA

El Plan de SQA identifica los productos a evaluar, los criterios a utilizar, los estándares a aplicar, las actividades a realizar y las técnicas a utilizar en cada caso.

8.2.1. Planificación de la Calidad

Se definieron varias actividades para el aseguramiento de la calidad del producto y de documentos relacionados al proyecto. Las mismas se realizaron para todas las áreas del proyecto, ya sea en la fase de investigación al inicio del proyecto, como en la planificación y realización de las distintas actividades de ingeniería de software. Estas actividades se encuentran documentadas en el [Anexo 6 - Plan de Calidad](#).

En el Plan de SQA que se encuentra en el [Anexo 7 - Plan de SQA](#), previamente mencionado, se definieron los estándares que se utilizaron para los documentos y la codificación. En el mismo también se indicaron las revisiones y validaciones del producto, y de algunos de los documentos más relevantes, describiendo qué se realizó.

Por último, se definieron las pruebas que se realizaron, cómo reportarlas y qué acciones correctivas se debieron tomar.

8.2.2. Estándares

Para el proyecto se define la utilización de diversos estándares dependiendo del entregable a realizar. A continuación, se listan los diferentes estándares definidos:

Estándares de documentación

Se define que se deberán usar los estándares de documentación 302, 303, 306 publicados por Universidad ORT Uruguay [35] [36] [37].

Los documentos que fueron realizados bajo estos estándares fueron el Plan de Proyecto, el Plan de Calidad, el Plan de SQA, el Plan de SCM, el ESRE, el Documento de Arquitectura y el presente documento.

Estándares de codificación

Se realizó una guía de buenas prácticas de codificación que se debió usar como estándar para toda la etapa de desarrollo de la aplicación. Esta está dividida en 3 grandes áreas: practicas generales, prácticas para el desarrollo en C# y prácticas para el desarrollo de código Java en Android. Estas se realizaron con el principal objetivo de lograr un código que cumpla con los principales estándares de la industria y que, al momento de que se continúe el desarrollo por parte de un tercero, pueda adaptarse fácilmente al mismo.

A continuación, se detallan los principales objetivos y algunas características del estándar definido. [\(Ver el Anexo 7 - Plan de SQA para más información\)](#)

El objetivo de la guía es:

- Estandarizar el desarrollo
- Establecer reglas.
- Hacer el código fácil de leer
- Facilitar la reutilización
- Lograr escalabilidad en el código [38]

Las características principales identificadas son:

- Evitar repetirse
- Código Simple
- Evitar Código Basura
- Nombres nemotécnicos
- Responsabilidad Única

- Uso de Interfaces
- Try-Catch

Para las diferentes tecnologías, se definió cómo se debe comentar el código y cuáles son las mejores prácticas para cada tecnología, siguiendo las guías correspondientes [39].

8.2.3. Revisiones

Dada la importancia de muchos de los procesos de ingeniería y los aspectos de seguridad de la aplicación, se realizaron actividades de revisión para asegurar que los pasos a seguir son los correctos y, de esta forma, minimizar los errores y correcciones que podían haber implicado la realización incorrecta de estas actividades.

En esta sección se detalla cómo se revisaron algunas de las características del producto, cuándo fueron revisadas y cuál fue el resultado de la revisión. Las características del producto revisadas fueron:

- Plan inicial del desarrollo – Adecuación
- Especificación de requerimientos – Correctitud
- Aplicación Móvil – Usabilidad
- Documento de arquitectura – Seguridad
- Documento de arquitectura - Completitud

Plan inicial de desarrollo - Adecuación

Al no ser un equipo con experiencia en el desarrollo de aplicaciones móviles, se decidió antes de comenzar con el mismo, realizar una reunión con un docente de la Universidad, Ignacio Valle. Los puntos más destacados de la reunión fueron:

- Características esenciales para el desarrollo de la aplicación. Esto incluye características de la arquitectura, como ser la separación de la interfaz de usuario por módulo, y la estructura necesaria para la comunicación desde la aplicación hacia afuera de la misma.
- Elementos de seguridad y cómo encararlos, por ejemplo, hacer el enfoque del lado del servidor y no tanto en la aplicación, o cómo realizar los servicios para que sean más seguros. Para esto se recomendó asignar a una persona que se enfoque en investigar qué es necesario para cumplir con la misma.
- Interfaz del menú. Dado que la decisión de cómo se muestra el menú en la aplicación tiene un impacto muy fuerte en el desarrollo, se plantearon varias opciones para saber cuál es la más eficiente y usable.
- La arquitectura del sistema

ESRE - Correctitud

La especificación de requerimientos fue también revisada por la tutora del proyecto, quien evaluó aspectos relacionados a la correctitud de los mismos antes de ser presentados a la contraparte del cliente.

Aplicación móvil - Usabilidad

Al finalizar las primeras 2 evoluciones de desarrollo, ya se contaba con una aplicación funcional, pero con una interfaz solamente basada en los bosquejos realizados al momento de definir los requerimientos. La usabilidad, al ser una de las restricciones más relevantes, y dado que una aplicación funcional, pero poco entendible para el usuario, es virtualmente inservible, se decidió consultar a un experto (Daniel Mordeki) para que revise la interfaz.

Los resultados obtenidos de la reunión implicaron un rediseño absoluto de la interfaz de usuario de la aplicación. Los cambios sugeridos y realizados se comentarán más adelante en el capítulo.

Documento de arquitectura - Seguridad

Siguiendo los consejos anteriores, se seleccionó a un integrante del grupo para investigar e implementar la seguridad del sistema.

Se investigaron varias herramientas que se adapten a las tecnologías y se idearon varias soluciones para el problema. Por ejemplo, el cifrado de contraseña en la base de datos, cifrado de todas las comunicaciones entre la aplicación y el servidor, y manejo de sesión de usuario.

El resultado fue:

- No se vio necesario la implementación de SSL, dado que afecta la performance e impacta directamente en el requerimiento no funcional, pues no se está trabajando con información sumamente sensible.
- Utilizar una sesión que se termine por *time out* o inactividad.
- Utilizar la contraseña de los usuarios cifrada en el inicio de sesión.

Documento de arquitectura - Completitud

Una vez finalizado el diseño general del sistema, habiendo tenido en cuenta los requerimientos no funcionales, existió una nueva reunión con Ignacio Valle para revisar la solución de la arquitectura propuesta.

Se encontraron algunos puntos de mejora que no afectaron a la solución propuesta. El principal fue identificar un módulo que sea común a todos para poder reutilizar código, dado que la solución presentada separaba cada módulo de forma independiente.

8.2.4. Validaciones

Los principales entregables del producto y productos derivados del proyecto fueron validados por expertos o por el cliente, usando reuniones como metodología de validación, o envió de documentos con las decisiones correspondientes.

Requerimientos funcionales

Luego de haber realizado el análisis de los requerimientos y generado un documento con una especificación a alto nivel de los mismos, se validó el mismo con el Sub Gerente General. Luego de especificados los requerimientos en un ESRE, estos fueron validados con el *Sponsor* del proyecto. Esto se realizó enviando una copia del ESRE por correo electrónico. Para cada requerimiento validado, se indica quién y cuándo lo validó y, en caso de que el mismo fue modificado, se vuelve a validar.

Requerimientos no funcionales

Los requerimientos no funcionales, dado la importancia de los mismos, fueron validados por varias personas con distintos roles dentro de la empresa. Esto se hizo mediante reuniones presenciales. Estas personas fueron: un Desarrollador, un Experto en base de datos, el responsable de las comunicaciones y el Gerente del Área de Sistemas, todos pertenecientes a la empresa.

Se realizó de esta manera para poder cubrir con todas las restricciones posibles que los interesados tienen sobre el proyecto.

Los requerimientos identificados y especificados fueron validados con los asistentes de la reunión posteriormente.

Interfaz

Luego de haber realizado los cambios sugeridos en la revisión de la interfaz, se realizó un documento con todos los cambios y las justificaciones de los mismos.

El documento fue enviado a Daniel Mordeki para su validación. La devolución fue muy positiva, pero de todos modos se encontraron algunos detalles a mejorar que fueron debidamente aplicados.

Seguridad

Dada la solución propuesta de seguridad, la misma fue presentada al referente en seguridad y redes dentro de la empresa (el Administrador de Redes), con el objetivo de validar la misma. Esto fue en el marco de una reunión informal a partir de una de las visitas a la empresa.

El referente estuvo de acuerdo con la mayoría de las decisiones, pero remarcó la necesidad de usar TLS/SSL al menos para el *log in* de la aplicación, dado que la información implicada en la operación es considerada muy sensible.

Finalmente, a raíz de la validación, se implementó la comunicación utilizando TLS/SSL entre la aplicación móvil y el *backend*. Esto se realizó inclusive para todas las comunicaciones, dado que no afectó el rendimiento. Luego de implementado, se realizó una validación final con el referente en seguridad y redes de la empresa, quien aprobó lo realizado hasta el momento en la materia.

La Aplicación

Cuando el desarrollo de un módulo se encontraba en su etapa final, el mismo fue presentado a los interesados del cliente pertenecientes a ese módulo particular, en formato de reunión, en la cual se mostraba el avance obtenido. De esta forma, se pudo validar lo realizado. Esto se decidió hacerlo antes de finalizar el desarrollo de forma que, en caso de surgir cambios, ya sea porque los requerimientos no eran exactamente los esperados o porque se haya omitido parte de la lógica del negocio por un requerimiento que no fue bien comprendido, poder trabajar en los cambios antes de terminar con la etapa de desarrollo de la evolución.

8.2.5. Revisiones de documentos y actividades

Para algunas de las actividades requeridas para el proyecto se realizaron revisiones de la planificación, de forma de asegurar que las decisiones elegidas eran las correctas. A continuación, se detallan las actividades que se revisaron y sus resultados.

Plan inicial de calidad

Dado que ninguno de los integrantes tenía experiencia en lo que implicaba realizar un plan de calidad y hacer un seguimiento del mismo, luego de realizarse una primera versión del Plan de la Calidad y de SQA, existió una etapa de revisión con un experto en la materia. Aprovechando el conocimiento del proyecto, se le pidió a la Tutora que fuese quien revise el mismo. A continuación, se detallan los resultados:

- Se realizó un plan muy genérico que, si bien es útil para el proyecto, no sirve para atacar los potenciales problemas del proyecto en particular.
- Las revisiones y las validaciones no fueron suficientes y contaban con el problema anterior de no ser específicas.
- Se sugirió realizar una tabla de comparación entre los requerimientos funcionales y los problemas que se presentarían a la empresa sobre el funcionamiento incorrecto, para determinar su nivel de criticidad y así poder priorizar en los críticos para asegurar que su función se cumpla de acuerdo a lo esperado.

SCM

La realización de este proyecto involucró un manejo de distinta información dada la naturaleza del cliente, la cantidad de personal y de interesados en el proyecto, y del ciclo de vida seleccionado, que implicó al menos un documento por actividad de cada evolución. Por lo tanto, se decidió realizar una revisión de las tecnologías elegidas como repositorio y de la estructura de carpetas para asegurar que las decisiones elegidas eran las correctas y, en caso de no serlo, obtener nuevas opciones para los elementos de configuración.

Revisiones de ORTs^f

Existieron 3 etapas formales de revisión académica realizadas por la universidad. Estas consistieron en realizar una presentación a un revisor de ORTs^f que no contaba con conocimiento alguno sobre el proyecto. Al final de las mismas, se dio una devolución de cómo vieron el proyecto. A continuación, se listan las oportunidades de mejoras principales que encontraron para el proyecto para cada revisión. Las revisiones completas se encuentran en el [Anexo 8 -Informe de Primera Revisión](#), [Anexo 9 -Informe de Segunda Revisión](#), [Anexo 10 -Informe de Tercera Revisión](#).

- Definir cuáles son los módulos a desarrollar, sus interesados y su prioridad.
- Reunirse con el cliente para mejorar la especificación de los requerimientos no funcionales.
- Mejorar la descripción del problema por parte del equipo.

8.3. Usabilidad

Se decidió enfocar parte del trabajo en cumplir con los requisitos de usabilidad, siendo considerada uno de los factores más importantes dentro de la calidad del producto de software.

8.3.1. Trabajo de interfaz de usuario

A partir de la reunión realizada con Daniel Mordeki para revisar y validar la interfaz de usuario de la aplicación móvil desarrollada para el proyecto, se obtuvieron varias sugerencias de cambios y correcciones. La mayoría de ellas estaban relacionadas a la alineación de componentes e información, y cómo mejorar la presentación. Todas ellas se pueden apreciar en los resultados de la reunión en el [Anexo 11 - Evolución de la Interfaz de Usuario](#), así como las mejoras realizadas a la aplicación luego de la misma, teniendo en cuenta los puntos mencionados.

Al finalizar la reunión, Daniel Mordeki propuso 3 reglas básicas que se deben seguir para tener una interfaz de usuario aceptable:

Alineación: mantener los distintos componentes alineados tanto horizontal como verticalmente. De lo contrario el ojo lo detecta como desprolijo.

Empujar contenido y funcionalidad hacia arriba: Siempre que se pueda, mostrarle la información al usuario lo antes posible. Cuantos menos *clicks* deba ejecutar, mejor.

Agrupación visual (representar jerarquías de familias): Distinguir los componentes hijos, padres y hermanos. Que se note qué componente se encuentra incluido en otro, y qué componentes son estructuralmente similares entre sí (pero distintos en contenido).

Como último resultado de la reunión, se muestran uno de los principales cambios realizados en la interfaz y su correspondiente justificación:

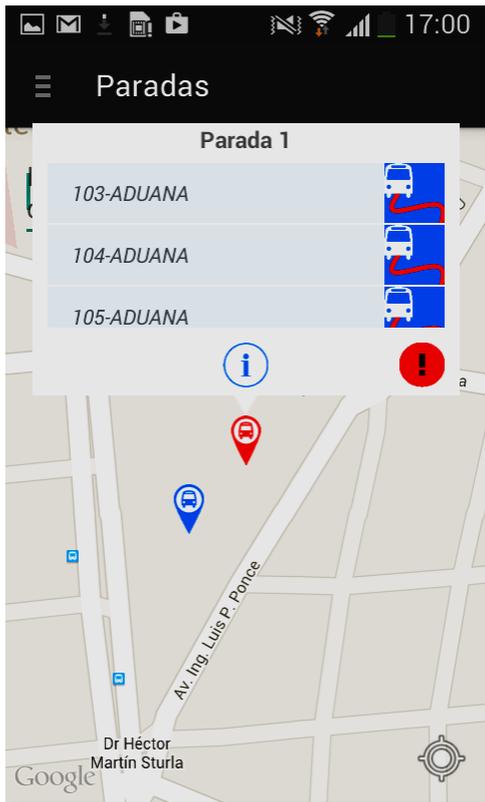


Ilustración 8-1 Información de una parada antes del cambio

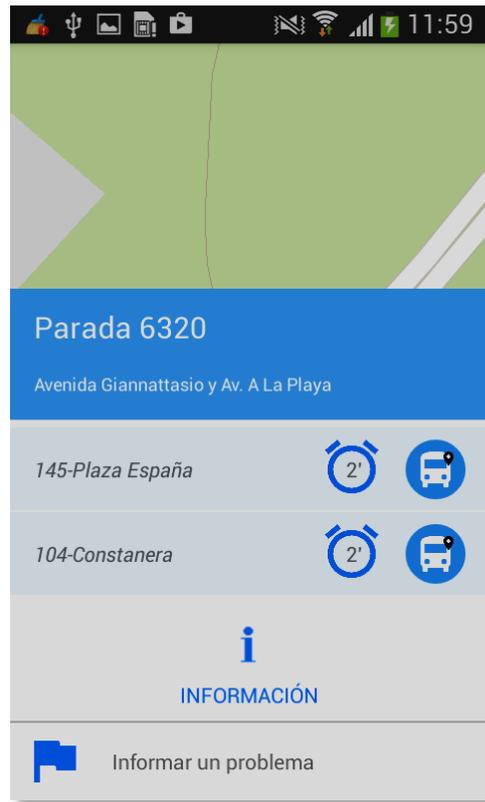


Ilustración 8-2 Información de una parada después del cambio

- El *popup* del marcador tuvo un cambio radical, siguiendo la forma en que Google muestra la información de los marcadores en su aplicación mapa.
- Se cambia la imagen del traqueo de ómnibus  por una más representativa y con un nuevo diseño .
- Se explicita el botón de información  mostrándolo de la forma en que lo hace y recomienda Google.
- El botón de reporte de error , el cual no quedaba muy explícito, se muestra de la forma en que Google usa para que el usuario reporte un problema.
- Se agrega un efecto de despliegue al seleccionar y deseleccionar un marcador, para que el mostrar y ocultar la información del marcador (y el componente de búsqueda del mapa) sea más animado (similar al que se usa en la aplicación Mapas de Google).
- Se “trae información hacia arriba”, permitiendo ver en esta vista el primer ómnibus que se acerca a la parada por cada recorrido. Esto se muestra primero con una indicación de carga (mientras se obtiene la información del servidor) y

luego se muestra al usuario cuántos minutos faltan para que llegue el ómnibus:

de  a .

- Toda información que debe ser cargada, tanto en esta vista como en otras a lo largo de la aplicación móvil, tiene una notificación hacia el usuario por medio de sonido (una pequeña alarma) y una breve vibración del dispositivo para avisarle que la información terminó de cargar. Dichas notificaciones no existían en la versión de la aplicación presentada a Daniel Mordeki en la reunión. Las mismas fueron agregadas a raíz de que los usuarios estarán la mayoría del tiempo utilizando la aplicación en la calle o sobre el ómnibus, y no pueden estar constantemente mirando el dispositivo para saber si terminó de cargar la información. Una pequeña alerta, mediante sonido y/o vibración, permite al usuario prestar atención a otros elementos importantes de su contexto (por ejemplo, ver si el ómnibus ya se está acercando a la parada), y continuar el uso de la aplicación cuando la carga realmente finalizó.
- Se permite minimizar la información del marcador, mostrando así la información de cabecera (número de la parada y su dirección) y un acceso directo a la función de “cómo ir en ómnibus”.

8.3.2. Aseguramiento de Usabilidad

Para asegurar que la aplicación cumpla con estándares de usabilidad, se basó en la guía creada por Jakob Nielsen [40]. La misma cuenta con 10 principios generales para el diseño de interfaces de usuario.

Los resultados detallados de las respuestas a los 10 principios se encuentran en el [Anexo 12 - Análisis Heurísticas de Nielsen](#). El resultado fue positivo dado que todos los principios fueron tomados en cuenta.

8.4. Testing

8.4.1. Planificación de la prueba

El objetivo de las pruebas en el proyecto fue cubrir todos los casos esenciales de la aplicación. Para esto se realizaron distintos tipos de pruebas que se describen a continuación.

Pruebas Unitarias:

Las pruebas unitarias las realizó el encargado del desarrollo móvil, generando pruebas automáticas para poder ser ejecutadas al final de cada evolución, y de esta forma comprobar que no se haya introducido un nuevo incidente luego del desarrollo de

nuevas funcionalidades. Estas pruebas solo se realizaron para el *backend* de la aplicación.

Las pruebas unitarias se implementaron utilizando el *framework* NUnit.

Pruebas de Integración:

Estas pruebas se realizaron para verificar la correcta integración entre el *backend* y la aplicación móvil.

Para estas pruebas se utilizó la técnica de pruebas funcionales o de caja negra, esto significa que se probó el comportamiento observable exterior del problema, desde la perspectiva del usuario.

Se estudió el comportamiento a partir de entradas y salidas como lo indica el diagrama



Ilustración 8-3 Ejemplo gráfico de forma de ejecución de prueba

Para estas se crearon planillas para cada evolución, en donde existe una página de inicio en donde están los resultados de ejecución de las pruebas y en las siguientes paginas las pruebas realizadas. Para cada prueba se encuentran detallados sus pasos y los resultados de su ejecución. Los datos requeridos para los casos de prueba se pueden apreciar en la siguiente ilustración.

ID	G-CP1				
Responsable	Andrés				
Fecha de Prueba	10/10/2015				
Caso de uso	G-CU1				
Objetivo	Probar que un usuario dado de alta previamente en el sistema pueda loguearse correctamente.				
Precondición	Que haya dado de alta en el sistema un usuario con número de funcionario "111111", contraseña "a123", nombre "Nombre1" y Apellido "Apellido1".				

Paso	Descripción	Resultado esperado	Resultado Obtenido	Resultado (Éxito o Error)	Ticket de Jira Asociado
1	Ejecutar la aplicación en el dispositivo móvil.	Se abre la aplicación	Se abre la aplicación	Éxito	
2	Se ingresa en el campo "Número de Funcionario" el valor "111111" y en el campo "Contraseña" el valor "a123" y finalmente se presiona el botón de "Iniciar Sesión".	Se accede a la sección principal de la aplicación, donde se ve en pantalla: "Nombre1" y "Apellido1" como datos del usuario logueado, y los menues correspondientes al Rol del usuario.	Se accede a la sección principal de la aplicación, donde se ve en pantalla: "Nombre1" y "Apellido1" como datos del usuario logueado, y los menues correspondientes al Rol del usuario.	Éxito	

Ilustración 8-4 Ejemplo de prueba de integración ejecutado

Pruebas de sistema:

Las pruebas de Sistema buscan evaluar el sistema en su entorno de operación, por esto requirieron acordar con el cliente la infraestructura necesaria y los requisitos para su realización.

Si bien se planificaron, no se pudieron realizar debido a restricciones planteadas por el cliente.

Durante las pruebas de carga y eficiencia, se intentó simular lo más cercano posible un entorno de operación típico. Se configuró el *backend* en el servidor físico donde a futuro se va a implantar el sistema y, además, se configuró la herramienta de testeo para que simule una comunicación mediante 3G, tal como se comunicaría un dispositivo móvil. Si bien esto no suplanta una prueba de sistema, dado que estas no se pudieron realizar, fue lo más cercano que se logró.

Pruebas de seguridad:

El objetivo de estas pruebas es comprobar que lo implementado en cuanto a seguridad está funcionando como se esperaba. Esto es que el encriptado de la contraseña se mantenga por más que el mensaje sea interceptado, los controles de la sesión funcionen según lo esperado y que las comunicaciones viajen mediante un canal seguro TLS/SSL.

Para esto se realizó una prueba en conjunto con el Administrador de Redes de la empresa, que consistió en la interceptación de paquetes de datos utilizando la herramienta Wireshark y comprobando la correcta utilización de TLS/SSL. En la siguiente ilustración 8.5 se puede ver un ejemplo de paquete enviado desde el *backend* hasta la aplicación móvil, donde los datos se muestran encriptados.

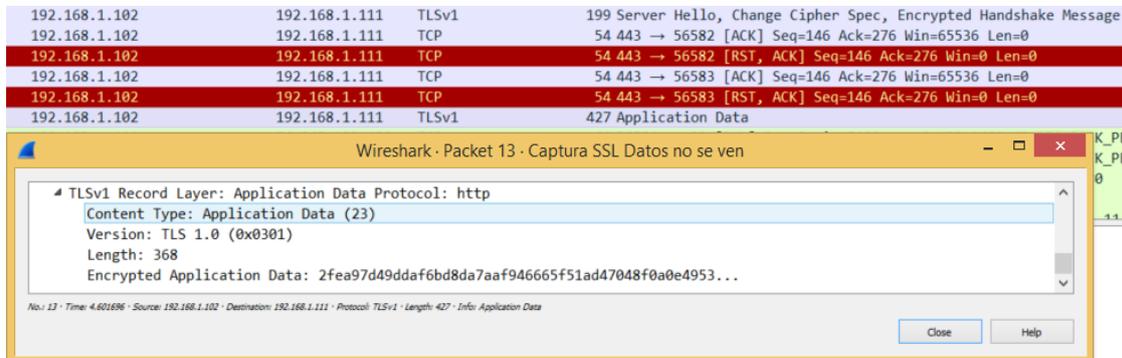


Ilustración 8-5 Paquete cifrado con TLS/SSL

Además, se realizaron *tests* para comprobar el correcto funcionamiento de la sesión de los usuarios y del encriptado de la contraseña.

La prueba completa se encuentra en el [Anexo 17 - Prueba de Seguridad](#).

Criterios de resultados de prueba

Los criterios que pueden ser asignados a los resultados de prueba en un Caso de Prueba son los siguientes:

- Exitosa: El resultado de la prueba corresponde a lo esperado.
- Errónea: El resultado de la prueba es completamente incorrecto o impide continuar el uso del sistema.

En caso de que la ejecución de un caso de prueba presente errores, tanto críticos como no críticos, este debe ser ingresado como un incidente en Jira.

En la Ilustración 8.5 se puede apreciar cuál es curso a seguir en caso de encontrar un error en el sistema.

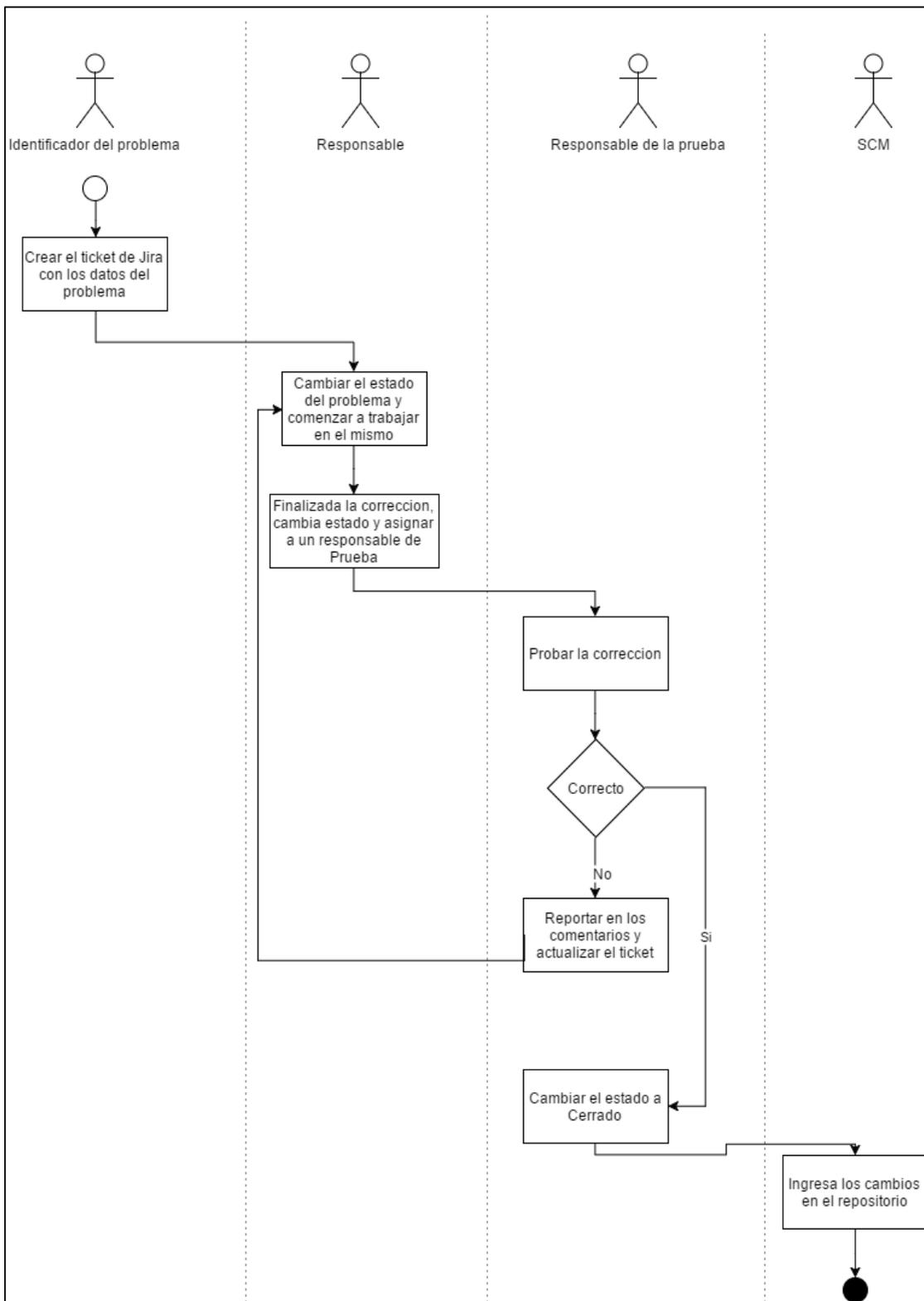


Ilustración 8-6 Diagrama de actividad de detección y corrección de prueba

8.4.2. Ejecución y resultados

Se definieron 2 premisas fundamentales para la ejecución de las pruebas: tanto la ejecución de las mismas como su especificación, sin importar qué tipo de pruebas fueron, siempre fueron realizadas por otra persona que no sea quien desarrolló.

Para reportar los errores y hacerles seguimiento, se utilizó la herramienta de Atassilan, Jira, ya que es una herramienta fácil de utilizar e intuitiva, en la cual se pueden visualizar reportes.

Al registrar un problema en la herramienta, se indicó la prioridad del mismo, el responsable y el estado del problema. También un nombre descriptivo, y una descripción detallada del problema. En muchos casos, se incluyeron imágenes para aclarar el problema.

La herramienta permitió realizar comentarios que fueron utilizados para agregar sugerencias para resolución del problema y para comentar sobre avances en el mismo.

Se tomaron 3 niveles de severidad adaptando el campo Prioridad de la herramienta:

- *High*: cuando el problema hace que el sistema se caiga o impida el uso de las demás funcionalidades.
- *Medium*: cuando el requerimiento no funciona o no cumple con la funcionalidad básica esperada para el mismo
- *Low*: cuando el problema refiere a un comportamiento errático de la aplicación, pero no implica que el requerimiento cumpla con su funcionalidad. También se usa cuando existen problemas con la interfaz gráfica.

Los problemas tienen 3 estados:

- **Por Hacer**: Este estado corresponde a los problemas que surgieron y todavía no se empezaron, o problemas que se han vuelto a abrir en caso de que se vuelvan a encontrar errores.
- **En Progreso**: Este estado indica que se está trabajando en el problema.

- Cerrado: Los problemas solo se podrán cerrar luego de que el mismo haya sido probado.

Ilustración 8-7 Ejemplo de un error reportado en Jira

Resultados de pruebas

Las pruebas, tanto las unitarias como las de integración, fueron realizadas por evolución.

Un ejemplo de resultado de ejecutar pruebas unitarias para la primera evaluación se puede ver en la siguiente ilustración.

Ilustración 8-8 Ejemplo de corrida de prueba unitaria

Un ejemplo de resultado de ejecutar pruebas de integración de un módulo del sistema se aprecia en la tabla.

Estadísticas de la ejecución	
Pruebas Ejecutadas	26

Pruebas Sin Ejecutar	0
Cantidad de pruebas exitosas	20
Cantidad de pruebas NO exitosas	6
Porcentaje de pruebas exitosas	78%

Tabla 8-1 Ejemplo de corrida de prueba de integración

Se ejecutaron 430 pruebas y el resultado obtenido fue de 348 exitosas y 82 No Exitosas. De esta forma, la cantidad de No Exitosas representa el 19% del total de las pruebas.

A partir de las pruebas, se encontraron 60 defectos que fueron ingresados a Jira. La cantidad de defectos fue menor que la cantidad de errores debido a que algunos errores estaban diseñados para probar lo mismo, pero de diferente forma. Al solucionar el problema, se solucionaban todos los errores relacionados.

8.5. Herramientas

Para el correcto desarrollo del proyecto, se definieron herramientas que sirvieron de apoyo para el aseguramiento de la calidad del producto.

- Procesador de texto - Microsoft office Word, Excel y Google Docs.
- Reporte de Incidentes - Jira, Atassilan [41]

8.6. Métricas de Producto

Errores encontrados

Esta métrica muestra el porcentaje de pruebas exitosas con respecto a las pruebas no exitosas

La cantidad de pruebas realizadas fueron 430, de las cuales 82 fueron Pruebas NO exitosas, y 60 Errores encontrados (tickets ingresados a Jira).

En la *Ilustración 9.8* se puede ver el porcentaje de pruebas exitosas y las no exitosas

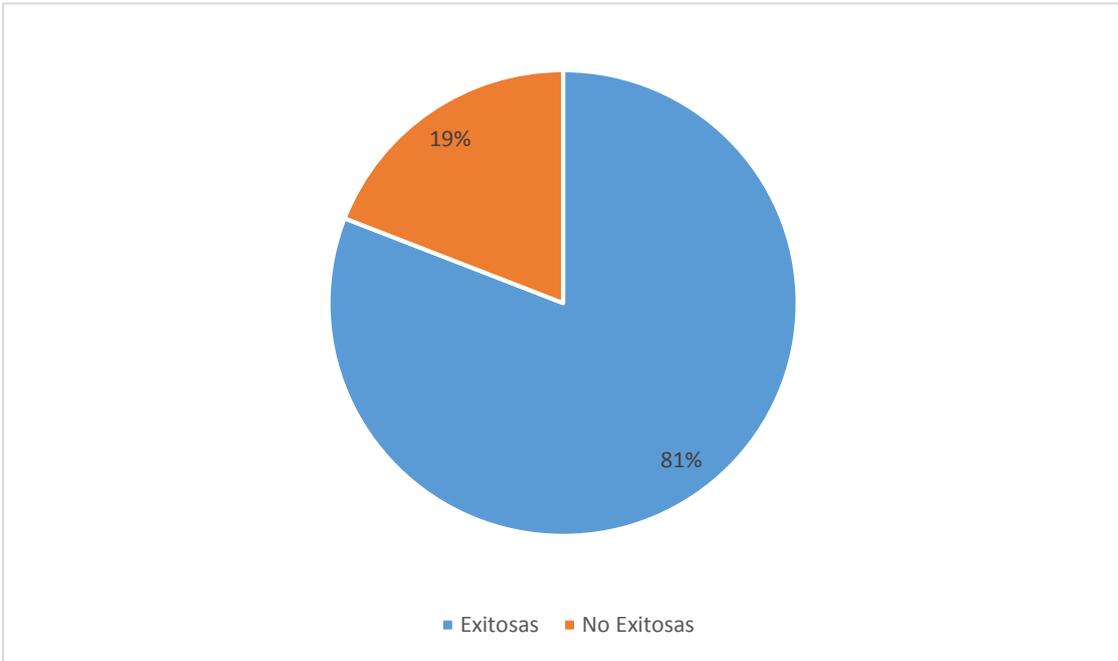


Gráfico 8-1 Porcentaje pruebas exitosas vs no exitosas

Errores encontrados por prueba

Esta métrica indica cuántas pruebas fueron necesarias para encontrar un error.

Analizando pruebas por error encontrado, se llegó a la siguiente ecuación:

$$\frac{\text{TotalDePruebas}}{\text{ErroresEncontrados}}$$

Ecuación 8-1 Pruebas por error encontrado

De esta forma, se concluyó que se necesitaron 7 pruebas para encontrar cada error registrado.

Errores Reportados vs Errores Corregidos

Esta métrica muestra la relación entre los defectos encontrados en la aplicación gracias a las pruebas realizadas y los defectos corregidos. La ilustración muestra la relación, la cual fue extraída de Jira

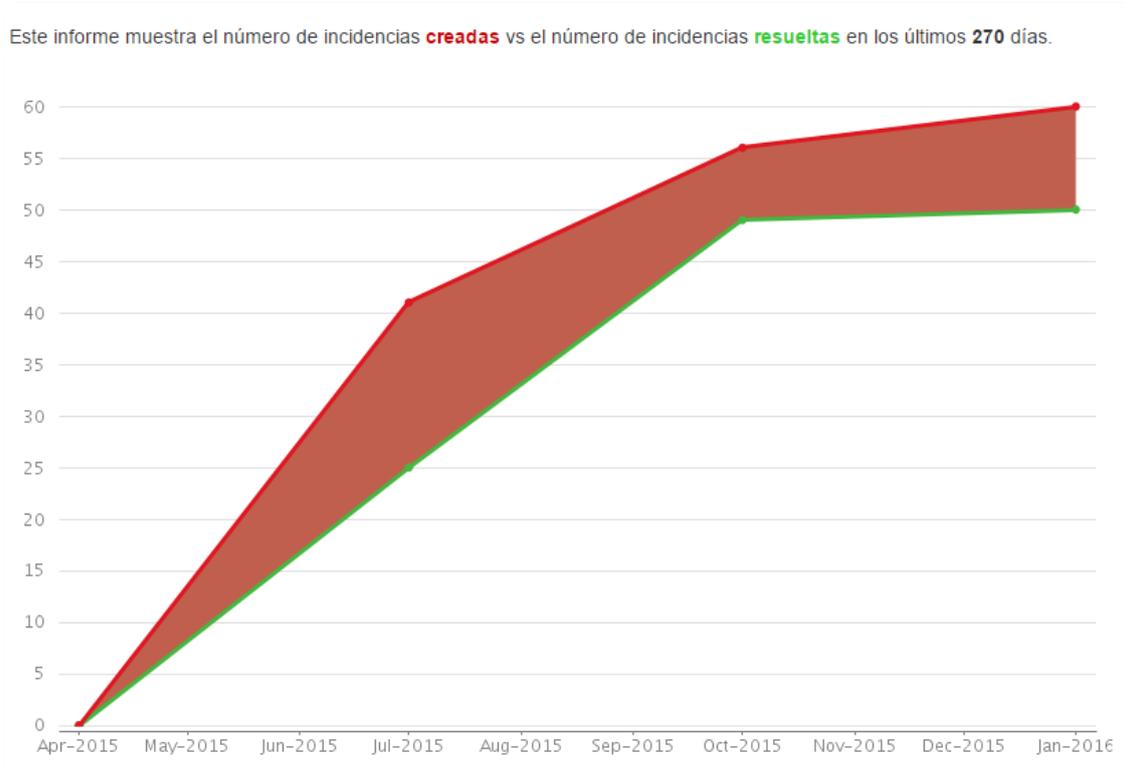


Ilustración 8-9 Errores reportados vs corregidos

De la gráfica, se puede apreciar que no todos los defectos encontrados pudieron ser corregidos antes de finalizada la etapa de desarrollo del proyecto. Existen 10 defectos que no se han solucionado.

Incidentes totales por producto

Esta métrica muestra la relación de la cantidad total de incidentes por producto.

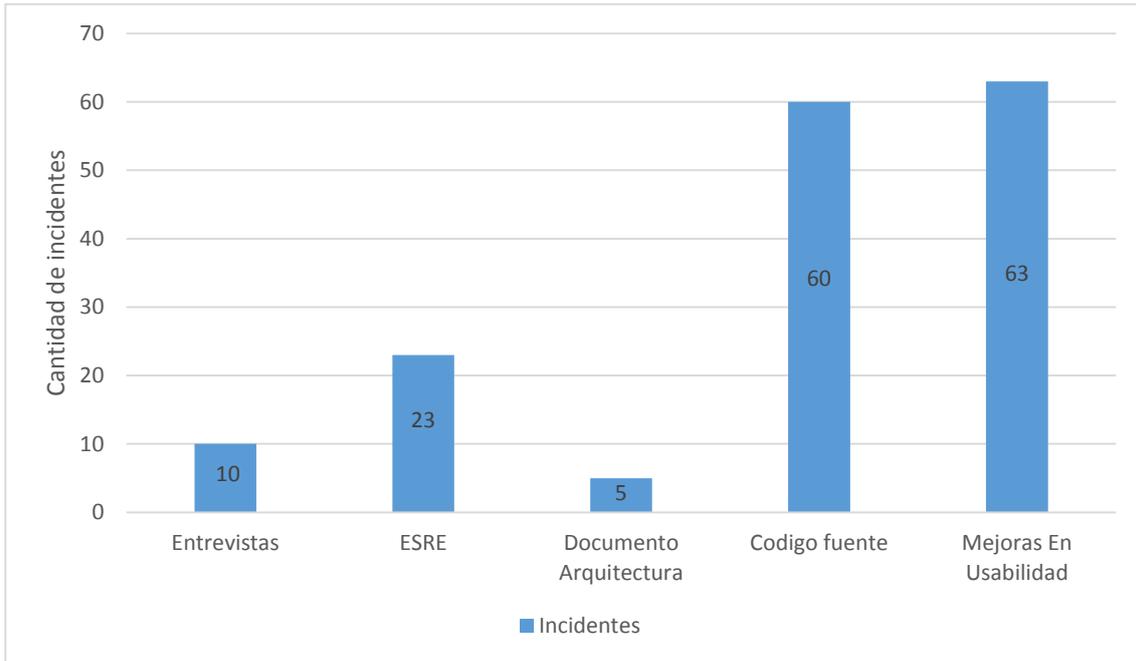


Gráfico 8-2 incidencias por producto

En la gráfica se puede apreciar que en todos los documentos y productos se han encontrado incidentes. Los datos de la columna de “Mejoras en usabilidad” se obtuvieron a partir de la revisión con D. Mordeki, que representan todos los cambios a la interfaz de la aplicación. Estos no se toman como defecto de código, ya que las funcionalidades eran correctas, pero no estaban cumpliendo con los estándares de usabilidad.

Incidentes corregidos por producto

El objetivo de esta métrica es mostrar qué porcentaje de incidentes encontrados por producto fueron resueltos.

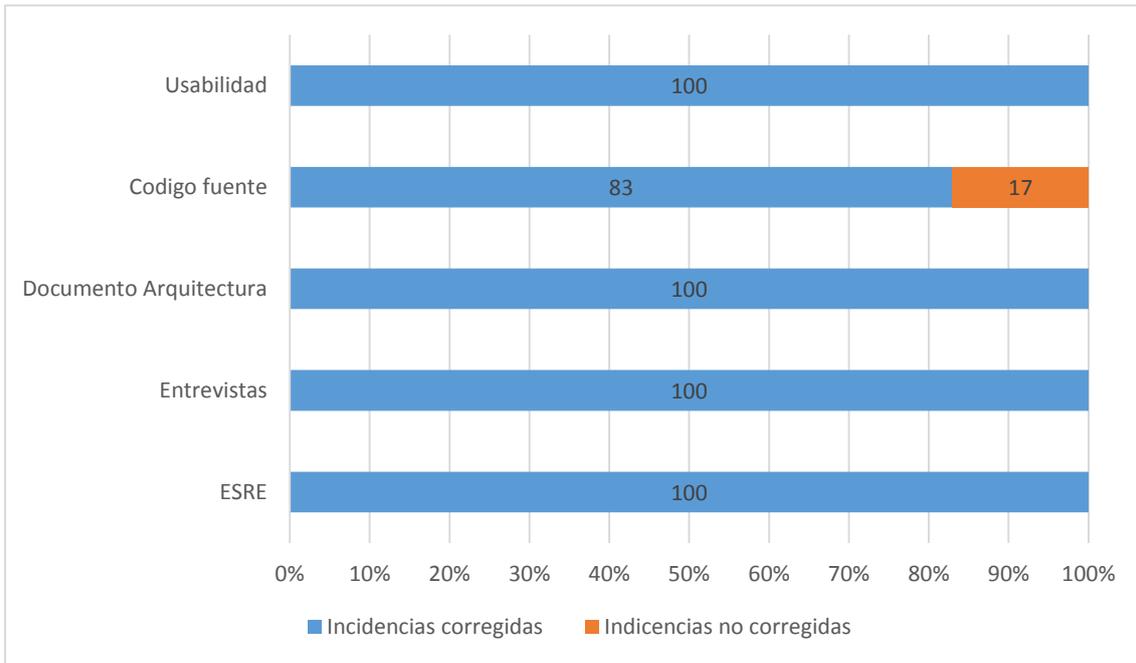


Gráfico 8-3 Porcentaje de incidentes corregidos

De la gráfica anterior se puede obtener que, a diferencia de los incidentes encontrados en el código fuente, todos fueron debidamente corregidos

8.7. Conclusiones y lecciones aprendidas

El esfuerzo en actividades de aseguramiento y control de la calidad permitió identificar 161 defectos a lo largo del proyecto relacionados a todos los productos. De los errores identificados, el 7% no fueron corregidos, lo que permite confirmar la utilidad de planificar y realizar actividades de aseguramiento de la calidad.

Gracias a las encuestas y reuniones realizadas con los diferentes interesados, se lograron validar y corregir los requerimientos obtenidos.

En las actividades de revisión de la arquitectura se logró conseguir la arquitectura modular y extensible que se buscó.

Otro aspecto positivo fueron las revisiones de usabilidad que resultaron en un rediseño de la aplicación, con diferencias notables.

9. Gestión de la configuración

En este capítulo se describe la gestión de la configuración del software a lo largo del proyecto. Para ver más detalle sobre los puntos descritos a continuación, ver el [Anexo 16 - Plan SCM](#).

9.1. Identificación de los elementos de la configuración

Se identificaron 4 tipos distintos de elementos de configuración del software (ECS), los cuales se desprenden del proceso de Ingeniería y de los procesos de apoyo. Los mismos se listan a continuación:

- Documentos de trabajo: Son documentos que sirven para el trabajo de los integrantes pero que no serán entregados para su evaluación.
- Documentos formales: Son documentos que necesitan más formalidad y revisiones ya que serán entregados para su evaluación.
- Multimedia: Son fotos, videos o audio registrados durante el transcurso del proyecto que es necesario tener disponibles para futura referencia.
- Código Fuente: Son todos los elementos que comprenden al código fuente, sea del *frontend* o *backend* del sistema.

9.2. Selección de herramientas

Debido a la naturaleza del proyecto, la elección de las herramientas se hizo en base a las categorías de archivos que se describieron en el apartado anterior

Para los elementos de configuración relativos a los **Documentos de trabajo**, se buscaron herramientas que cuenten con las siguientes características:

- Historial de versiones por documento.
- Posibilidad de revertir a una versión anterior de un documento.
- Sincronización con la nube con el fin de acceder desde cualquier lugar y tener un respaldo robusto.
- Compatibilidad con formato de archivos de Microsoft Office.

Para los ECS relativos a los **Documentos formales**, se buscaron herramientas que cuenten con las siguientes características:

- Historial de versiones por documento.

- Posibilidad de revertir a una versión anterior de un documento.
- Sincronización con la nube con el fin de acceder desde cualquier lugar y tener un respaldo robusto.
- Compatibilidad con formato de archivos de Microsoft Office.
- Edición simultánea online del documento por parte de los integrantes del equipo.

Para los ECS relativos a los archivos **Multimedia**, se buscaron herramientas que cuenten con las siguientes características:

- Historial de versiones por archivo.
- Sincronización con la nube con el fin de acceder desde cualquier lugar y tener un respaldo robusto.

Los ECS relativos a los archivos **Código Fuente**, se buscaron herramientas que cuenten con las siguientes características:

- El código sea privado para personas externas del equipo de proyecto.
- Acceso para al menos 3 desarrolladores.
- Preferentemente gratuito.
- Posibilidad de recuperar versiones anteriores.
- Integración con Microsoft Visual Studio para el *backend*.
- Integración con Android Studio para el *frontend*.

Finalmente, luego de analizar diversas herramientas (se pueden ver los análisis en el [Anexo 16 - Plan SCM](#)), se decidió utilizar las que se muestran en la siguiente tabla:

Elemento de configuración	Herramienta
Documentos de trabajo	Google Drive
Documentos formales	Microsoft OneDrive
Multimedia	Google Drive
Código Fuente	Bitbucket para el <i>frontend</i> Microsoft TFS para el <i>backend</i>

Tabla 9-1 Herramientas seleccionadas

Cabe aclarar que en un principio se había seleccionado para el *backend*, BitBucket como repositorio y SourceTree como herramienta de versionado, opción que luego de probarla en los primeros pasos del desarrollo presentó inconvenientes al momento de integrar versiones de los distintos miembros del equipo, por lo que se decidió cambiar por TFS.

9.3. Gestión de repositorio

Para los repositorios de la documentación, se tuvieron 2 grandes consideraciones. Por un lado, se separó en carpetas y se versionó los documentos que debieron de ser mostrados y enviados para su aprobación, separando las versiones enviadas a aprobar de las ya aprobadas.

Por otro lado, debido al ciclo de vida elegido, algunos documentos, principalmente el ESRE, se vieron modificados en cada evolución. Por lo cual, se crearon carpetas separando archivos específicos de cada una de estas. En la siguiente ilustración se puede ver la estructura de directorios utilizada.

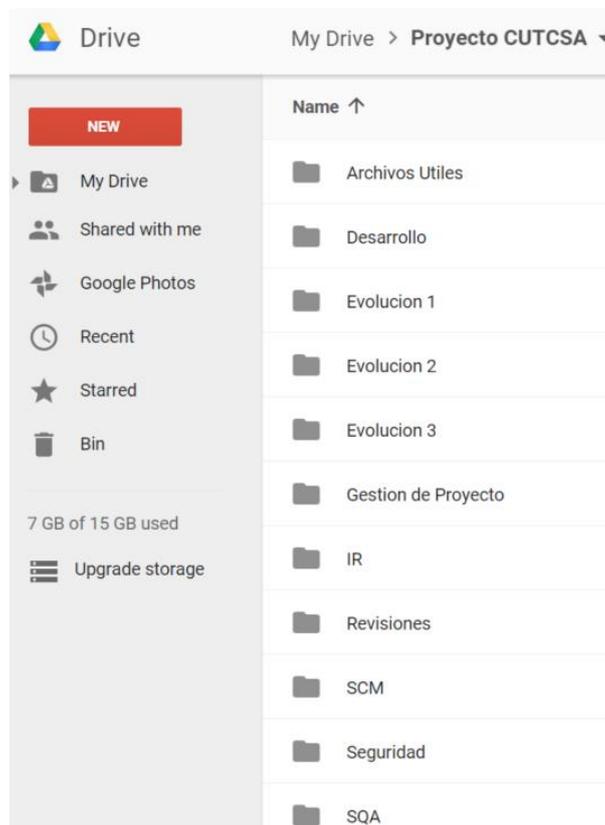


Ilustración 9-1 Estructura directorios documentos.

Para los repositorios del código fuente, debido a que en general se desarrolló de forma desacoplada, se decidió no utilizar ramificaciones del código, y además utilizar solamente dos repositorios, uno para la aplicación Android y otro para el *backend*, es decir, no utilizar múltiples repositorios por cada sub componente del sistema.

En la siguiente ilustración se puede visualizar el repositorio en Bitbucket de la aplicación móvil, en donde se pueden ver algunos *commits* realizados por integrantes del equipo.

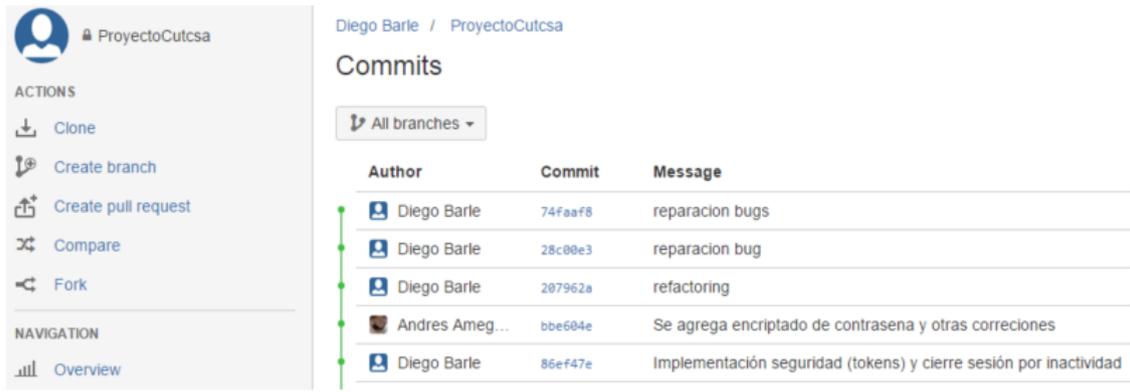


Ilustración 9-2 Repositorio aplicación Android.

9.4. Release de versiones

A lo largo del desarrollo, en los momentos en los que se alcanzaron ciertos hitos, o se desarrollaron funcionalidades específicas, se liberaron versiones para que se pueda probar con ellas o usarlas para validar con el cliente.

El estándar utilizado para nombrar los componentes liberados del *backend* fue: BXX.YY, donde XX representa un número de versión mayor, el cual cambia cuando se agregan grandes funcionalidades, YY representa un número de versión menor, el cual cambia a medida que se agregaron o arreglaron funcionalidades menores.

Análogamente para los componentes liberados del *frontend*, se utiliza el siguiente estándar: FXX.YY.

Dado que el sistema está compuesto por ambos componentes, se definió una tabla en la cual se especificaban las versiones de *backend* y *frontend* que juntas, empaquetadas, formaban una versión de *release*, para la cual se utilizó el siguiente estándar: VXX, siendo XX un número. En la siguiente tabla se puede observar un ejemplo:

Release	Versión backend	Versión frontend
V01	B02.05	F02.06
V02	B02.08	F03.01

Tabla 9-2 Ejemplo de mapeo de Release.

9.5. Conclusiones

En términos generales, las herramientas seleccionadas fueron usadas satisfactoriamente y cumplieron su función de forma eficiente.

Al comienzo, no fue fácil la utilización de las herramientas de versionado para algunos integrantes, por lo que, mediante la generación de guías de configuración y utilización de las herramientas de versionado, se logró facilitar la utilización de los mismos.

Se intentó utilizar el mismo tipo de repositorio tanto para la aplicación móvil como para el *backend*, lo cual no resultó ya que en el *backend* no se logró utilizar correctamente. Por otro lado, el equipo se dio cuenta de que no era realmente necesario utilizar el mismo repositorio y, por ende, decidió pasar a utilizar en el *backend* otro con mucho mejor integración con el entorno de desarrollo usado, el cual facilitó enormemente la tarea de respaldar y versionar.

De lo que se concluye que hubiera sido una buena práctica, además de haber seleccionado el repositorio y herramienta de versionado, haber podido realizar alguna pequeña prueba de utilización del mismo en conjunto con el entorno de desarrollo usado.

10. Conclusiones

El proyecto tuvo como resultado una solución de software que involucra una aplicación móvil Android nativa y un *backend*, que interactúan con los sistemas del cliente para obtener la información utilizada por la aplicación. Requirió un esfuerzo de 1983 horas registradas, con un tamaño de 295 puntos de función para las funcionalidades desarrolladas.

10.1. Conclusiones generales

A continuación, se muestran las conclusiones obtenidas a partir del análisis del cumplimiento de los objetivos planteados. Las citas utilizadas en esta sección del capítulo fueron realizadas por el Sub Gerente General de C.U.T.C.S.A., las que fueron extraídas de una entrevista realizada para presentar el producto y evaluar los resultados obtenidos. El resultado completo sobre su evaluación del proyecto se puede ver en el [Anexo 21 - Reunión final con el Sub Gerente General de C.U.T.C.S.A.](#)

Objetivo: El objetivo del producto será ayudar a mejorar procesos de comunicación y manejo de información para diferentes áreas de la empresa C.U.T.C.S.A., así como también mejorar la eficiencia y eficacia de tareas mediante la digitalización de información y procesos.

Resultado: Con el uso de una aplicación móvil, se va a permitir a los empleados de la empresa consultar información y tenerla disponible en lugares donde hoy en día no podían acceder a la misma. Además, se les brinda una herramienta de trabajo que ayuda a agilizar varios procesos. En la entrevista al Sub Gerente General, él brindó su opinión al respecto, y su respuesta fue la siguiente:

“(...) permitirá mejorar la gestión de áreas muy sensibles de la empresa desde el punto de vista de la gestión de la flota y de la información a nuestros clientes (externos e internos). (...) Esa mejora de gestión resultará de contar con información más completa y en tiempos muy útiles para simplificar las tareas del personal, por lo que entiendo que es muy positivo desde el punto de vista del aumento de la eficiencia en el cumplimiento de las tareas habituales.”

Objetivo: El objetivo del proyecto es crear una aplicación Android que sirva para mejorar y facilitar el proceso de trabajo de al menos 2 roles de empleados distintos dentro de la empresa.

Resultado: Se logró alcanzar este objetivo dado que la aplicación Android desarrollada abarca a 3 roles de empleados de la empresa. Estos son:

- Inspectores: La aplicación les brinda 14 funcionalidades.
- Guardas/Conductores/Conductores-Cobradores: La aplicación les brinda 8 funcionalidades.
- Largadores: La aplicación les brinda 10 funcionalidades.

Cabe destacar que hay 7 funcionalidades generales para todos los roles actuales y futuros que tenga la aplicación.

Objetivo: Crear una solución tecnológica sostenible, que facilite su análisis posterior y extensibilidad a nuevos procesos del negocio y actores dentro de la organización, basándose en la premisa que la aplicación seguirá siendo desarrollada en un futuro por el cliente.

Resultado: El sistema desarrollado cuenta con una arquitectura modular, la cual permite la extensión del mismo, y una completitud de documentos que ayudan a la comprensión de la solución implementada.

Objetivo: Lograr cumplir los compromisos asumidos con el cliente y este quede satisfecho con el producto final.

Resultado: El común denominador de las diversas validaciones realizadas a distintas partes interesadas dentro de la empresa fue la aprobación de lo realizado durante el proyecto por el equipo. El equipo está convencido de que el contacto constante con el cliente ayudó a que esto haya sido de esta manera. A continuación, se puede ver la opinión del Sub Gerente General al respecto.

“(...) a partir de su puesta en producción será inevitable que crezca integrando otras áreas y funciones. Pero en esta instancia el alcance es incluso mayor al imaginado en primera instancia, especialmente por su interacción y complementación con la aplicación que desarrollamos (...).”

Objetivo: Aportar innovación al sistema de trabajo de la empresa.

Resultado: La empresa solo contaba con aplicaciones de escritorio, y muchas comunicaciones se realizaban mediante papeles o llamados telefónicos. Con el sistema desarrollado, se aporta innovación en procesos a través de la utilización de nuevas tecnologías y sustituyendo o complementando los métodos de comunicación, adicionando la posibilidad de utilizar un móvil como tal. De esta manera, también se permite realizar tareas de forma digital en lugares donde la empresa no puede estar presente (en la calle, terminales y sobre los ómnibus).

A continuación, se puede ver la opinión del Sub Gerente General al respecto:

“(…) Conozco la forma de trabajo y las herramientas de gestión en muchas ciudades del mundo. Este proyecto nos posiciona en vanguardia, especialmente teniendo en cuenta las condiciones de nuestro tránsito. Poder integrar información sobre el estado del mismo y generar alarmas será útil, no solo para nuestro personal y clientes, sino también para el resto de los componentes del sistema (peatones, automovilistas, autoridades, etc.).”

Objetivo: Utilizar las habilidades técnicas adquiridas en la carrera para poder llevar a cabo este proyecto tecnológico.

Resultado: En las áreas de Ingeniería de requerimientos, Desarrollo, Arquitectura, Pruebas y Gestión de proyectos se hizo un fuerte uso de lo aprendido hasta el momento en la carrera. No siendo de esta forma en las áreas de SQA y SCM.

Puntualmente, fue un gran desafío la definición del Plan de SQA y sus actividades relacionadas, manejar las distintas versiones de un software utilizando una herramienta, y la investigación e implementación de los requisitos de seguridad.

De esta manera, concluimos que hubiese ayudado mucho haber cursado antes o durante el proyecto materias electivas relacionadas a la calidad y seguridad.

Objetivo: Obtener experiencia real en gestión de proyectos y en nuevas tecnologías para generar una base de lecciones aprendidas y de esta forma buscar crecer a un nivel profesional.

Resultado: Con respecto a la gestión del proyecto, puntualmente, existió una mejora importante en relación a la precisión del cronograma y esfuerzo a dedicar, gracias a los análisis de puntos de función y registro de horas. Y, con respecto a la gestión de riesgos, aportó de forma significativa para prevenir posibles problemas que pudieran generar retrasos, a pesar de que algunas decisiones para mitigarlos, como fue el plan de entrenamiento, generó una ligera pérdida de tiempo.

A lo largo del desarrollo el equipo se enfrentó a tecnologías de las cuales no se tenía conocimiento previo, como ser: NFC, TLS/SSL, códigos QR, y que finalmente podemos decir que logramos adoptar.

Objetivo: Aprender a gestionar el proyecto en todas sus fases y desafíos. Debido a que el proyecto tendrá ciclos completos de desarrollo, se deberán poder gestionar todas las fases de la ingeniería de software para poder lograr un producto de calidad para el cliente.

Resultado: Efectivamente, la gestión del proyecto fue un desafío, mejorando poco a poco a medida que se obtenía experiencia en ello. Para la misma, y de forma satisfactoria, se apoyó en los distintos análisis para aumentar su precisión del esfuerzo a dedicar, y también realizar una organización completa de actividades, riesgos y entregables.

En resumen, podemos concluir que todos los objetivos planteados para el proyecto se han cumplido de acuerdo a lo esperado y en algunos casos superando las expectativas del *Sponsor* del proyecto.

10.2. Líneas de trabajo para el futuro

Como se comentó durante el documento, el sistema es modular, cuyo objetivo es seguir expandiéndose en funcionalidad para los roles realizados, y creación de nuevos roles. Es por dicha razón que se realizaron distintos documentos para ser entregados a la empresa, y con ellos se pueda entender las decisiones tomadas para un desarrollo futuro de la aplicación. A su vez, se realizó un sistema expandible desde el punto de vista del código, respetando los estándares utilizados por la empresa, y del mundo del desarrollo en sí.

Queda por corregir algunos defectos detallados en los tickets pendientes, los cuales fueron documentados y entregados al cliente.

Con las decisiones tomadas, documentos entregados y de la solución final en sí, se obtuvo por parte del *Sponsor* (Manuel Ares) la siguiente respuesta. Su opinión completa se puede apreciar en el [Anexo 23 – Reunión final con Sponsor](#)

“Conforme con la aplicación, de la cual se continuará con la ambición que tiene la empresa de querer más (...), haciendo cosas porque al proyecto le veo mucho futuro y muy interesante para la explotación de la empresa (...)”.

11. Bibliografía

- [1] Google, «Developers,» [En línea]. Available: <http://developer.android.com/guide/components/activities.html>. [Último acceso: marzo 2016].
- [2] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/Advanced_Encryption_Standard. [Último acceso: febrero 2016].
- [3] Wikimedia Foundation, «Wikipedia,» febrero 2016. [En línea]. Available: https://en.wikipedia.org/wiki/Application_programming_interface.
- [4] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/C%C3%B3digo_QR. [Último acceso: febrero 2016].
- [5] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/Graphics_Interchange_Format. [Último acceso: febrero 2016].
- [6] Google, «Google Developers,» [En línea]. Available: <https://developers.google.com/places/>. [Último acceso: Febrero 2016].
- [7] Intendencia Municipal de Montevideo, «Tránsito y transporte,» [En línea]. Available: <http://www.montevideo.gub.uy/transito-y-transporte/observatorio-de-transito>. [Último acceso: febrero 2016].
- [8] json.org, «Introducing JSON,» [En línea]. Available: <http://json.org/json-es.html>. [Último acceso: febrero 2016].
- [9] Google, «Developers,» [En línea]. Available: <http://developer.android.com/intl/es/guide/topics/ui/declaring-layout.html>. [Último acceso: febrero 2016].

- [10] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://en.wikipedia.org/wiki/Near_field_communication. [Último acceso: febrero 2016].
- [11] Uruguay, Universidad ORT, «Laboratorio ORT Software Factory,» 2016. [En línea]. Available: <http://fi.ort.edu.uy/2013/17/objetivos.html>.
- [12] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/M%C3%A9trica_de_punto_funci%C3%B3n. [Último acceso: febrero 2016].
- [13] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://es.wikipedia.org/wiki/Secure_Hash_Algorithm. [Último acceso: febrero 2016].
- [14] «Wikipedia,» Wikimedia Foundation, [En línea]. Available: https://en.wikipedia.org/wiki/Information_technology. [Último acceso: febrero 2016].
- [15] Microsoft Corporation, «TechNet,» [En línea]. Available: https://es.wikipedia.org/wiki/Wikipedia:Acerca_de. [Último acceso: febrero 2016].
- [16] Wikimedia Foundation, «Wikipedia,» [En línea]. Available: https://en.wikipedia.org/wiki/User_interface. [Último acceso: febrero 2016].
- [17] Object Management Group, «Unified Modeling Language (UML) Resource Page,» [En línea]. Available: http://www.omg.org/gettingstarted/what_is_uml.htm. [Último acceso: febrero 2016].
- [18] Uruguay, Universidad ORT, «Laboratorio ORT Software Factory,» [En línea]. Available: <http://fi.ort.edu.uy/laboratorio-ort-software-factory>. [Último acceso: 2016].
- [19] C.U.T.C.S.A., «cutcsa,» 2014. [En línea]. Available: <http://www.cutcsa.com.uy/index.php?oid=9>. [Último acceso: 2016].

- [20] R. Pressman, Ingeniería del Software: un enfoque práctico, Mcgraw Hill Editoria, 2004.
- [21] ISO/IEC, 25010:2011 Systems and software Quality Requirements and Evaluation (SQuaRE), 2011.
- [22] Google, "Material Design," [Online]. Available: <http://www.google.com/design/spec/material-design/introduction.html>. [Accessed Agosto 2015].
- [23] IDC, «Smartphone OS Market Share, 2015 Q2,» 2015. [En línea]. Available: <http://www.idc.com/prodserv/smartphone-os-market-share.jsp#>. [Último acceso: Junio 2015].
- [24] P. C. R. K. L. Bass, Software Architecture in Practice. 3ra ed., Massachusetts: Addison-Wesley Professional, 2012.
- [25] D. Mordeki, Guía práctica de Usabilidad web miro y entiendo, Montevideo: Mastergraf SRL, 2012.
- [26] M. Baron, Probability and Statistics for Computer Scientists. 2da. ed., New York: Chapman and Hall/CRC, 2013.
- [27] everac99, «Estimando usuarios concurrentes,» Junio 2009. [En línea]. Available: <https://everac99.wordpress.com/2009/06/04/estimando-usuarios-concurrentes/>. [Último acceso: Diciembre 2015].
- [28] Y. L. Y. H. Y. Xiaoyun Wang, "Finding Collisions in the Full SHA-1," 2005. [Online]. Available: <https://www.iacr.org/archive/crypto2005/36210017/36210017.pdf>. [Accessed Octubre 2015].
- [29] C. Palmer, «Google Online Security Blog,» 5 Setiembre 2014. [En línea]. Available: <https://googleonlinesecurity.blogspot.com.uy/2014/09/gradually-sunset-sha-1.html>. [Último acceso: Octubre 2015].

- [30] M. A. Alvarez, «desarrolloweb,» 2014. [En línea]. Available: <http://www.desarrolloweb.com/articulos/ventajas-inconvenientes-apirest-desarrollo.html>. [Último acceso: Julio 2015].
- [31] WhatsApp Inc, «WhatsApp,» [En línea]. Available: <https://www.whatsapp.com/?l=es>. [Último acceso: febrero 2015].
- [32] marcucio, «Manage It,» [En línea]. Available: <https://manageitapp.com/>. [Último acceso: febrero 2016].
- [33] Microsoft Corporation, «Microsoft Project 2010,» [En línea]. Available: Microsoft Corporation. [Último acceso: febrero 2015].
- [34] Google, «Google Drive,» [En línea]. Available: https://www.google.com/drive/?_ga=1.238786496.1535449302.1444857050. [Último acceso: febrero 2016].
- [35] Uruguay, Universidad ORT, «Universidad ORT Uruaguay,» 2015. [En línea]. Available: <http://www.ort.edu.uy/facs/pdf/documento303carreras cortasfacs.pdf>. [Último acceso: 2015].
- [36] Uruguay, Universidad ORT, «Universidad ORT,» 2015. [En línea]. Available: <http://www.ort.edu.uy/facs/pdf/documento302carreras cortasfacs.pdf>. [Último acceso: 2015].
- [37] Uruguay, Universidad ORT, «Universidad ORT Uruguay,» 2015. [En línea]. Available: <http://www.ort.edu.uy/varios/pdf/documento306.pdf>. [Último acceso: 2015].
- [38] I. Canarias, «Slideshare,» Febrero 2012. [En línea]. Available: <http://es.slideshare.net/ikercanarias/buenas-prcticas-para-la-construccin-de-software>. [Último acceso: Julio 2015].
- [39] Microsoft, «Microsoft Developer Network,» [En línea]. Available: [https://msdn.microsoft.com/es-es/library/hh156542\(v=vs.110\).aspx](https://msdn.microsoft.com/es-es/library/hh156542(v=vs.110).aspx). [Último acceso: 2015].

- [40] Agesic, «Agesic,» [En línea]. Available: http://www.agesic.gub.uy/innovaportal/file/549/1/Capitulo_2_Usabilidad_v1_0.pdf. [Último acceso: Enero 2016].
- [41] Atlassian, «Atlassian Software,» 2015. [En línea]. Available: https://es.atlassian.com/software/jira/?_mid=262765b02ed8e86f262bd2697e123351&gclid=Cj0KEQIAu9q2BRDq3MDbvOL1yaYBEiQAD6qoBgd9Jglzr-lxbf2Rp1aDI5Wq0HD31wnJ82N6_gKvVzMaAo6h8P8HAQ. [Último acceso: 2015].
- [42] Microsoft, «MSDN Microsoft,» 2015. [En línea]. Available: <https://msdn.microsoft.com/en-us/library/b2s063f7.aspx>. [Último acceso: 07 2015].
- [43] I. Canarias, «slideshare,» Febrero 2012. [En línea]. Available: <http://es.slideshare.net/ikercanarias/buenas-prcticas-para-la-construccin-de-software>. [Último acceso: 2015].
- [44] Microsoft, «Microsoft Developer Network,» 2015. [En línea]. Available: <https://msdn.microsoft.com/es-es/library/ff926074.aspx>. [Último acceso: 2015].
- [45] Oracle, «Oracle,» [En línea]. Available: <http://www.oracle.com/technetwork/articles/java/index-137868.html>. [Último acceso: 2015].
- [46] Medline Plus, «Medline Plus Informacion de salud de la biblioteca nacional de medicina,» 2015. [En línea]. Available: <https://www.nlm.nih.gov/medlineplus/spanish/ency/article/003387.htm>. [Último acceso: 11 2015].
- [47] Google, «Google Material design,» 2015. [En línea]. Available: www.google.com/design/spec/style/color.html#color-ui-color-application. [Último acceso: 11 2015].
- [48] Wikipedia, «Wikipedia,» 2015. [En línea]. Available: <https://es.wikipedia.org/wiki/Daltonismo>. [Último acceso: 11 2015].

[49] Microsoft Corporation, «TechNet,» [En línea]. Available: <https://technet.microsoft.com/en-us/library/cc784450%28v=ws.10%29.aspx>. [Último acceso: febrero 2016].

[50] Microsoft Corporation, «Microsoft Project 2010,» [En línea]. Available: <https://www.microsoft.com/latam/gobierno/productos/project2010.aspx>.

12. Anexos

12.1. Anexo 1 - Situación actual de la empresa

12.1.1. Situación actual de la empresa

12.1.1.1. Antecedentes

12.1.1.1.1. Sobre la Empresa

Misión

“Brindar a la población soluciones de transporte modernas, seguras, confiables y eficientes, teniendo como meta la satisfacción del cliente. Descubrir constantemente los requerimientos del mercado en materia de transporte y adaptarnos a ellos en forma rápida y eficiente. Realizar todo emprendimiento que agregue valor a nuestro negocio, permita optimizar el resultado para nuestros Accionistas y represente una contribución positiva para la Sociedad.”

Visión

“Ser una empresa de referencia permanente en el servicio de transporte colectivo, impulsora de los cambios que permitan brindar un servicio público de calidad, de personas para personas y donde el cliente debe ser lo primero.”

Valores

“Honestidad y lealtad, Compromiso –de todos con todos-, Respeto –servicio de personas para personas-, Vocación de servicio –responsabilidad social-, Profesionalismo – dignificar nuestra tarea-, Creatividad/Innovación –tecnología e ideas-.”

Estructura de la Empresa

La empresa está compuesta por alrededor de 1200 conductores, 1100 conductores cobradores, 1200 guardas, 300 controladores de servicios, 300 que mantienen la flota, y 600 directores, administradores y de servicios generales; así sumando casi 5000 trabajadores. Algunos de ellos son también propietarios (llegando a una suma de casi 3200) de casi 1200 sociedades de hecho (ómnibus y prestadoras de servicios) organizados en 122 recorridos.

Los empleados se organizan principalmente en 5 áreas operativas (Ver [ANEXO 1](#)).

En primer lugar, el **Área de Recursos Humanos y Administración** compuesta por el Tribunal Disciplinario, el Departamento de Recursos humanos, el Departamento Administrativo, y el Departamento Jurídico (reclamos laborales o de otras índoles, CUTCSA seguros). “Se encarga del manejo de los recursos humanos (selección, capacitación, evaluación del desempeño, administración de altas, bajas, movimientos, enfermedades, licencias, etc.).

En particular, el Departamento de Recursos Humanos involucra el ingreso del nuevo personal, coordinación de nuevos ingresos para realizar los módulos de instrucción, trámites y préstamos otorgados por el BROU, comprobantes de certificados de trabajo, duplicado de recibo de sueldos, planilla de trabajo, suscripción de licencias anual reglamentaria a gozar, cambios de puesto a otra categoría, control de desempeño (donde se convoca al personal de la empresa que comete una falta, de acuerdo con el Reglamento Disciplinario, a efectos de realizar los descargos correspondientes), e interrupciones de trabajo por enfermedad.

En el Departamento Administrativo se mantiene el archivo general de la empresa, se realiza el diseño de formularios, revistas, materiales internos (aquí también se encuentra el diseño gráfico), se mantiene el control de consumo de la empresa (electricidad, teléfono, etc.), se encarga de pagar tributos y comisiones, realiza vales de consumos de coches, y se firman liquidaciones de egreso y de choques.

La Secretaria General, que también se encuentra en esta área, se encarga de tareas de secretaría de directorio y gerencia, toma actas de directorio de la empresa, los cuales emiten un comunicado aprobación del cambio de puesto, etc.

En la Oficina de Despacho es donde se emite el libre tránsito para que puedan viajar gratuitamente en el transporte urbano. También se emite el padrón de propietarios que pueden votar y ser elegidos en elecciones internas; el padrón de propietarios que pueden asistir a asambleas; y realiza trámites ante la Intendencia de Montevideo y Canelones sobre transferencias de las unidades, cambio de titularidad, etc.

El **Área de Sistemas y Estudios Tecnológicos** compuesta por el Departamento de desarrollo y el Departamento de operaciones. “Tiene a su cargo todo el soporte informático de la Empresa, los sistemas operativos, la provisión y el mantenimiento del hardware, software y comunicaciones para el resto de las áreas y en particular, la implementación del proyecto de tecnología sobre los autobuses”. Las tareas correspondientes al departamento de operaciones es recibir información (esencialmente liquidaciones de sueldos y coches) fija, procesarlas y emitir información a departamentos, y realizar depósitos bancarios a los funcionarios.

El **Área de Transporte y Producción** compuesta por el Departamento de transporte, el Departamento de Producción, el Departamento de Investigación y Mercado y el Departamento de Desarrollo Social. “Se encarga de todo lo relacionado con el servicio de transporte. Incluye la planificación del servicio, la asignación del personal, la rotación de los servicios, el contralor del buen funcionamiento, la solución de contingencias, además de las tareas de investigación y estudios de mercado”.

El Departamento de Transporte realiza trabajos estadísticos que permiten adaptar servicios según la demanda, los cuales incluyen información a través de planilla, resultado de producción (diarios, mensuales y anuales), horarios a realizar o ajustes mensuales; todo relacionado a resultados productivos de la empresa. A su vez se encarga de las comunicaciones y servicios que incluyen avisos que hay en los coches (recorridos, condiciones de viajes para escolares, pases libres nuevos, etc.); de avisos o comunicaciones internas (que van a carteleras de las líneas), que pueden incluir anuncios o novedades de la IM, o avisos al personal; y de avisos a los clientes. También se encarga del mantenimiento de terminales, que incluye guardia, limpieza, abastecimiento de suministros (toallas, papel higiénico, jabón, etc.). Y a su vez se encarga de los trámites y gestiones ante organizaciones públicas (IM, Ministerio de transporte, etc.).

La tarea más importante del Departamento de Producción es poner en funcionamiento el servicio según lo planificado. Esto implica organizar la red de horarios, turnos, asignar personal para los servicios, y controlar y fiscalizar para que dichos servicios se cumplan. En particular, las tareas de las inspecciones incluyen el control de boletería, cumplimiento de horario, control a los clientes (que se le haya expedido correctamente el boleto, que tenga el boleto correspondiente al viaje que está realizando, o saber si realmente pasó tarjeta o no, dado que a veces no se le imprime boleto), control del estado de la unidad (que se encuentre en condiciones, limpio, etc.), la fiscalización general del servicio, control de la presentación del personal (usando uniforme, trato con el cliente, que no deje clientes en la parada, etc.), registrar el comportamiento de la competencia, la regulación del servicio, ajustes a la planificación según los eventos e imprevistos, y el aprovechamiento del servicio a la hora pico para favorecer la mayor cantidad de pasajeros.

El **Área Comercial y Servicios** compuesta por Suministros, Taller Mecánico, Plantas Auxiliares, Dirección Técnica, Comisión de Seguridad, Recauchutaje y Gomería, Taller de Electricidad, Administración, Mantenimiento, Centro de Monitoreo, y Locales de Venta. “Se encarga del funcionamiento de todos los servicios comerciales de apoyo a la flota”.

Compone todos los predios de la empresa, donde se gestionan todos los talleres, servicios, mantenimiento, reparaciones (preventivo y correctivo), compra de repuestos tanto en plaza como en exterior (importaciones), gestionan los predios, estacionamiento, retiro de combustibles, estaciones de servicios propias, repuestos y depósito, y venta de boletos de estudiantes.

Área de Contabilidad y Finanzas, compuesta por la Comisión de Finanzas, el Departamento Contable, el Departamento Financiero y el Departamento de Recaudación. “Encargada del control de las dependencias de recaudación, y locales de comercialización de boletos. Tiene a su cargo la planificación financiera del negocio”.

El departamento de recaudación está compuesto por 7 oficinas de recaudación donde se trabajan 24 horas diarias (una de ellas está compuesta a su vez por un anexo, Oficina 3, que trabaja solo 7 horas diarias.), donde se trabajan 24 horas. A su vez hay una sala de recuento de dinero, y una sala de recuento de boletos las cuales trabajan en dos turnos.

Todas estas áreas se rigen bajo la Gerencia General, que a su vez se rige por el Directorio y comisión fiscal, y todos a su vez por la Asamblea General de Accionistas.

A su vez existen áreas de apoyo:

La **Auditoría interna** que se encarga “del control de los Activos y del cumplimiento de los procedimientos establecidos. Reporta directamente a la Gerencia General y a la Comisión de Asuntos Administrativos. Colabora con la mejora de la gestión de las sociedades propietarias”.

La **Asesoría Contable**, “responsable del cumplimiento de las normas contables en la registración y elaboración de los estados, así como en la liquidación de cuenta corriente de propietarios. Responsable también de las liquidaciones fiscales de las personas jurídicas involucradas. Asesora en materia económica, financiera y tributaria.”.

La **Asesoría Legal** que “asesora al Directorio y a los responsables de las diferentes áreas en toda la materia legal (laboral, seguridad social, derecho comercial, penal, etc.)”.

Y el **Departamento de Gestión Humana y Relaciones Públicas** que “se encarga de llevar adelante todas las acciones que la empresa decide llevar adelante para mejorar las condiciones de trabajo de los trabajadores. Por ejemplo, campañas de vacunación, charlas sobre problemas de salud (diabetes, hipertensión, drogadicción, antitabaco, etc.). Prepara material sobre estos temas para los boletines informativos, *mailing*, avisos. Acompaña a familiares de personas que pudieron haber sufrido daños importantes por algún accidente de tránsito o rapiñas. También es la persona que recibe las posibles denuncias por acoso sexual, laboral, y problemas de género. Depende de la Gerencia de Recursos Humanos”.

12.1.1.1.2. Historia de la Organización

Gestación de CUTCSA

En el año 1926 surge una agrupación de orden gremial llamado el Centro de Propietarios de Ómnibus. El cual contaba dentro de sus principales objetivos:

- establecer un servicio competente de ómnibus para el transporte de personas de la ciudad de Montevideo.
- Establecimiento de un servicio regular a efectos de que los pasajeros tengan la seguridad de transporte.
- Propiciar este medio de transporte por ser seguro, rápido, cómodo y confortable.
- Distribución de coches según la demanda de las distintas líneas.
- Permanente contacto con las autoridades municipales con el fin de pautar reglas y formas de trabajo.

Dentro de las primeras actividades se destaca la reglamentación del servicio de trasbordo en caso de accidente y/o rotura y el primer llamado de precios para la compra a granel de nafta.

Posteriormente y llegando el año 1928 comienzan a integrarse cooperativas por líneas con el fin de unificar las compras y reducir el costo de las mismas.

Resultaba imprescindible para competir con los tranvías, además de generar un frente para defender los intereses, ofrecer un mejor servicio al usuario, generando tranquilidad, seguridad, confianza y poder en caso de roturas, dar una solución al pasajero. De esta manera permisarios con vehículos considerados distinguidos por los pasajeros debido a los nombres que lucían, comenzaron a colocar en sus parabrisas el nombre de la cooperativa que integraban.

Durante el año 1927, las empresas tranviarias solicitan autorización para la importación de 15 autobuses para completar sus servicios lo que no es avalado por el Municipio.

Entre 1928 y 1933 eran 15 las cooperativas de autobuses integradas, siendo la primera de ellas "La cooperativa de ómnibus línea D" y la última de este proceso la "F norte". De esta manera distintos permisarios comienzan a realizar diferentes líneas llegando en 1930 a un grupo de doce de ellos a fundar la "Cooperativa de ómnibus Pando-Montevideo y derivados", que posteriormente se convirtiera en "Cooperativa de Ómnibus de Pando S.A." COPSA.

Fundación

Dada la presión de la lucha entre la empresa tranviaria y los propietarios de ómnibus, en 1936 el Municipio de Montevideo, exigió mediante el decreto N° 1178 que en un plazo máximo de un año todos los permisarios de autobuses se nuclearan en una sociedad con personería jurídica bajo amenaza de caducarles los permisos.

En ese mismo año una asamblea del Centro de Propietarios aprueba el proyecto de estatutos de CO.N.AU.S.A. (Cooperativa Nacional de Autobuses Sociedad Anónima) a efectos de contemplar lo establecido en el decreto anterior.

Pasados algunos meses, las autoridades municipales, mediante el decreto N° 1402 de fecha 27/04/1937 variaron las exigencias ya que para otorgar la concesión la Sociedad debía ser propietaria al menos del 75% de la totalidad de los autobuses que cumplían servicios en Montevideo. Solo dos días más tarde según el decreto N° 1401 de fecha 29/04/1937 se autoriza a La Sociedad Comercial de Montevideo a reemplazar los tranvías por *trolleybuses*.

El Centro de Propietarios aprueba los estatutos de la nueva sociedad anónima creada para cumplir con el decreto N° 1178 sustituyendo el nombre de CO.N.A.U.S.A. por el de C.U.T.C.S.A.

Finalmente, el 22 de diciembre de 1937, la Intendencia de Montevideo, promulgó el decreto N° 1750 otorgando a la Cooperativa Uruguaya de Transportes Colectivos S.A. - C.U.T.C.S.A.- la concesión por dieciséis años para la explotación de los servicios de transporte colectivo en la ciudad de Montevideo, en vehículos de tracción automotriz con cualquier combustible salvo energía eléctrica.

12.1.1.1.3. Proceso macro de la empresa

A diario se trasladan 650.000 personas en 11.100 viajes, recorriendo 280.000 kilómetros.

A partir de las entrevistas realizadas a varios jefes de departamentos y gerentes de áreas, obtuvimos la siguiente descripción de cada uno.

Departamento de Operaciones

Existen comunicaciones con Recursos Humanos quienes informan sobre novedades de los funcionarios (por ejemplo, faltas o suspensiones, horas extras, feriados trabajados), datos que ellos liquidan para ser usado en los depósitos bancarios.

Con el Departamento Contable, Departamento Financiero y Departamento Jurídico (incluido el Departamento de Recursos Humanos) se transmite información o datos a través de comunicados.

También varias cooperativas de consumo y emergencias médicas se comunican con este departamento, vía e-mail, informando los gastos o consumos de los funcionarios

generando los descuentos que impactarán en los haberes y así generar las liquidaciones de sueldos cuyo resultado se envía al Dpto. Financiero para depositarlo en las cajas de ahorros de la red de Cajeros Automáticos.

Además, los Departamentos Contable y de Recursos Humanos se reciben los descuentos a realizar por concepto de: préstamos del BROU, BHU, etc.

Se reciben las planillas de los horarios tanto planificados como los que realmente se hicieron.

Departamento de Recaudación

El proceso del departamento comienza desde el coche con el guarda/conductor cobrador tomando el dinero, boletos OCA, tickets y el pendrive que contiene la información de las ventas que se realizaron; y se lleva a cualquiera de las oficinas de recaudación las cuales contienen una recontadora de monedas, puestos de autogestión donde colocan el pendrive descrito anteriormente y digita todos los valores que aparece en la liquidación e importe/valores contado por el Guarda/conductor cobrador, y se guarda el dinero en bolsas especiales. Las mismas son llevadas a un lugar especial en la empresa, donde se recuenta el dinero bajo vigilancia de cámaras de video y audio. Luego se chequea el dinero contado con las liquidaciones que los Guardas/Conductores cobradores entregan. En caso de haber diferencias (excedentes o faltas), se le notifica al Guarda/Conductor cobrador que puede chequear dichas diferencias en los servicios de Autogestión. En caso de no estar de acuerdo, puede pedir para ver las grabaciones del recuento de dinero. Luego, se procede a la limpieza de los billetes (limpieza y separación de billetes marcados o de mal estado y de billetes en estado óptimo requerido por el Banco Central) dentro del Búnker, y finalmente, al cerrar la jornada, se procede a realizar el depósito bancario donde también se hace el movimiento del dinero a través de Prosegur.

Es fundamental la comunicación entre el Búnker con Tesorería y el depósito en los bancos.

También se realiza el control del personal, se le ofrece cambio de dinero y rollos de boletos para trabajar en el ómnibus.

En las oficinas se pagan los tickets de alimentación y cuenta corriente de los coches, por lo que existe comunicación con el departamento de Recursos Humanos por liquidaciones de horas de trabajo y novedades (funcionarios que se enferman, o faltan, etc.).

Departamento de Producción

Existe una unidad de asistencia rápida para el caso de los accidentes o choques y eso se informa vía papel, el cual termina en jurídica donde lo manejan a nivel de abogados, sector de reclamación, y seguro. A su vez, la persona (conductor que tuvo el choque) está obligado a hacer un trámite por separado para completar informe (no lo hace enseguida de sucedido el accidente). Ese informe es distinto al que realizó el inspector en el momento del choque (se hacen distintas preguntas), porque es distinto ver la situación en el momento que sucedió, a verlo más tarde “en frío”. El informe del inspector está orientado a ver cuál es la responsabilidad (o no) del personal o de la parte contraria en el accidente. El conductor, en el trámite que realiza aparte, está obligado a asistir a la empresa.

Cuando sucede un accidente, existen dos situaciones: si hubo o no heridos. En caso de haber heridos, el coche debe quedarse detenido. Inspección asiste al lugar, realiza el llamado y la toma de datos, y se espera las resoluciones de la justicia para saber si el coche puede volver a trabajar. Por parte del conductor, el mismo se comunica con la oficina de su línea y comienza el protocolo.

En el caso de no haber heridos, el inspector no va, y la policía no interviene. El conductor recaba los datos y debe asistir al departamento para realizar los trámites (descritos anteriormente). En el caso de que el coche no se haya roto severamente y se encuentre en condiciones de continuar, se continúa con el servicio. De todas formas, el conductor llama a la línea la cual le explica los pasos siguientes. Para ver el diagrama del proceso descrito, ver el [Anexo 2](#)

Se llama al departamento en casos de situaciones difíciles o graves.

El manejo y creación de horarios de las líneas se realiza artesanalmente. No se puede hacer mediante software porque hay pautas o características estructuradas, y no es parametrizable (cambia tantas veces de tiempo y de situaciones como el recorrido lo necesite). Lo que sí se realiza luego es transcribir en un programa todo lo que se hace a mano, y como resultado se genera un input que se carga en las máquinas de los coches y con el cual se realizan los controles y tareas en el Smart.

Los horarios se cambian por temporadas (demanda, situaciones particulares, elecciones, vacaciones, estación).

Se trata de mantener los horarios básicos de los funcionarios para que sea lo más constante posible.

La forma de trabajo de los coches en la diaria y el personal que diariamente va en los coches los asignan los listeros en el software preparado para ellos. Y esto sucede en cada oficina de línea, todos los días desde las 5 de la mañana hasta las 11:30hs. Después de las 12 tienen que presentar las listas del día siguiente.

La mayoría del personal efectivo en los coches diariamente retira el “cartón” (hoja y horarios del día siguiente). Dicho cartón es reusable porque representa los servicios del día y no cambia según el coche). El que está efectivo de un coche, trae el cartón del día anterior, y se le entrega el del día siguiente. De esa forma ya sabe qué horarios debe hacer al otro día. Aparte de eso, también existe una lista mensual en donde el efectivo del coche puede verificar qué horarios le va a tocar en todo el mes, y qué días tiene los descansos.

Luego existe la asignación diaria del personal de lista que debe ir a la oficina de recaudación (aprovecha en el momento que tiene que realizar la entrega de la recaudación) y verifica el listado de los horarios que se le asignó para el día siguiente.

Área de Recursos Humanos y Administración

El departamento de Recursos humanos atiende el ingreso del nuevo personal, desde su presentación (referencias laborales y personales, carné de salud, salud en condiciones aceptables, etc.). Al ser aceptado, pasa a los módulos de instrucción donde se le da una introducción sobre la empresa, tecnología a usar de forma teórica, práctica sobre los ómnibus, etc. Luego firma la licencia, se realiza el ingreso efectivo (el cual también firma), y se realiza la convocación para que los responsables estén al tanto de su disponibilidad para el trabajo. Se realizan trámites de préstamos, se emiten comprobantes de certificación de trabajo, se emiten duplicado de recibo de sueldos, y se mantiene la planilla de trabajo.

Compradores y vendedores de unidades que administra C.U.T.C.S.A. se presentan ante la Oficina de Despacho a efectos de realizar el Planteamiento Previo por el cual las partes declaran las condiciones de la Compra Venta que pretenden realizar, el que luego de completar diversos trámites internos se eleva a consideración del Directorio. Si el Planteamiento Previo es rechazado por el Directorio se informar a las partes involucradas. Si es aceptado continúa el trámite informándose al Escribano Público actuante para que el mismo proceda a realizar el correspondiente la Escritura de la operación la que se presenta en CUTCSA a efectos de ser aprobada por parte del Directorio.

Se responsabiliza del cierre de padrón de propietarios y se responsabiliza del cierre anual del padrón de propietarios y accionistas que pueden ir a las asambleas.

Departamento de Transporte

La comunicación al personal se realiza a través de carteleras y volantes (el último en caso de ser un tema muy importante).

La Comunicación hacia clientes a través de la mampara de los coches.

Se comunica con recursos humanos del cual mayormente se recibe de ellos las comunicaciones.

A oficina de despacho se le lleva toda la información de cada coche (tipo, propietario, trámites) lo cual se realiza vía teléfono (en caso de ser urgente), sino se actualiza la información formalmente.

Con Contaduría (incluye tesorería) se tramita el pago a empresas tercerizadas para limpieza y seguridad. A su vez se realizan control de facturas y se le comunica a contaduría.

Se comunica también con el Centro de Atención al Cliente (CAC), donde se comunica mucho hacia ellos por el tema de novedades (dado que CAC recibe muchas consultas y reclamos, y deben estar al tanto de dichas novedades). Una de las copias del comunicado que se le da a los coches, se ofrece al CAC. A su vez, el CAC consulta al departamento sobre situaciones particulares que no tienen clara la respuesta a dar a los usuarios.

Área Comercial y de Servicios

Los servicios pueden ser agendados, o si es urgente se pasa directamente por el taller. Primero el guardia, cuando llega el coche, ingresa su número y en caso de que haya lugar en el taller, lo deja entrar; sino queda en pista. Desde el taller tienen un monitor que les avisa que el coche ya se encuentra en planta, y permite consultar la historia del coche (repuestos que compró, mano de obra) como una especie de ficha clínica. En dicha historia puede agregar los datos nuevos y advertencias a los propietarios de servicios u otras reparaciones que quedarían pendientes.

En algunas ocasiones, el coche queda más de un día parado por reparaciones, y el propietario suele estar muy pendiente para que el mismo pueda volver en circulación. A él se le pregunta cuándo tiene el próximo servicio (trabajo en la calle) para saber si llega al turno o si debe suspenderse y avisar a transporte. Usualmente se sabe el tiempo que demora la reparación, sino se para, se quita del servicio, y al personal que iba a realizar el servicio se le asigna a otro coche para que siga trabajando. Para ver el diagrama del proceso descrito, ver el [Anexo 3](#)

El estacionamiento de un coche dentro de los predios lo realiza un funcionario asignado que los deja apretados unos con otros (por tema de espacio) y, a través de una logística, deja disponible los coches que deben salir antes por servicios.

Frente a un problema, se le pide al conductor, por ejemplo, que ofrezca con detalle (por escrito) sobre el reclamo de inconformidad de una reparación, y lo firma. Aquí se investiga:

Se pide informes al encargado de turno (y eventualmente a los obreros que trabajaron en el coche).

Van a los listados de los repuestos que compraron y mano de obra que se cobraron.

Se compara el informe del reclamo con el informe del capataz.

Se junta todos los elementos y se realiza una reunión (llamada comisión técnica) en la cual participan los capataces de las áreas, el principal de mecánica, el principal de electricidad, el principal de carrocería, el gerente comercial y de servicios, el jefe de compras, y entre todos definen y analizan el problema, y determinan dónde ocurrió la falla (se analiza la causa de la falla y se trata de resolver).

Por otro lado, esta área se comunica con el resto por temas particulares.

Con Informática por ventas de gasoil para contabilizar y cobrar.

Con Recursos Humanos: por asistencia o enfermedades, y temas de esas índoles.

Con Finanzas por facturaciones y reparaciones, importaciones. Los ingresos de facturación se ingresan al sistema (programa) contable y al sistema del departamento.

Con transporte por penalizaciones o citaciones de coches, advertencias a responsables de coches. La comunicación se realiza a través de comunicados, teléfono o mail porque no están tan dependientes uno del otro.

La carga de combustible se comunica con el sistema de ANCAP y con el sistema del departamento.

12.1.1.2. Problemática actual

12.1.1.2.1. Problemas

Dentro del Departamento de Producción, los mayores problemas surgen de factores externos ajenos a la empresa como por ejemplo tráfico y choques.

En el Departamento de Operaciones, a veces puede haber diferencias con las novedades de hora, entre la información que recibe este departamento y lo que dice el documento sobre lo que tiene que sumar.

Los problemas más importantes del Departamento de Recaudación son los cortes de luz, o desperfectos que originan que queden colgados los puestos de autogestión. De todas formas, Se ha ido solucionando y disminuido este tipo de problemas a medida que se fueron agregando generadores. La mayoría de los problemas son conocidos y se resuelven en el momento de forma rápida.

Para el caso del Área de Recursos Humanos y Administración, hay problemas para realizar correctamente el control de horario del personal.

12.1.1.2.2. Oportunidades de mejora

De las entrevistas pudimos concluir que existe una oportunidad de mejoras con respecto a la comunicación, tanto para pasaje de comunicados, pedir abastecimiento de insumos, etc. El diagrama de comunicación obtenida a partir de las entrevistas puede observarse en el [Anexo 4](#).

El resto de las oportunidades se separaron entre las distintas áreas:

Departamento de Operaciones

- A. Mejorar la información que tienen los clientes en general.
- B. El tener que llevar la información de los depósitos al banco en Diskette. El banco tiene su aplicación, pero el directorio en ese caso aún no lo ha validado.
- C. El proceso de las liquidaciones si bien está funcionando correctamente no es muy eficiente.

Departamento de Recaudación

- A. Hoy en día no existen documentos bases que describan las tareas o procesos que deben tener los funcionarios porque en estos últimos tiempos ha sido muy dinámico y han cambiado muchas cosas. De todas formas, se les hace llegar a los funcionarios a través de comunicados. Para aprender una tarea, hay un encargado que se les enseña, luego el funcionario practica y en una semana ya se encuentra en condiciones para trabajar de forma autónoma.

- B. Se mueve mucha información dentro de la oficina de recaudación en el depósito, control de rollos de boletos, control administrativo de horarios de personal que trabaja en las oficinas, y llamadas de problemas por máquinas de autogestión.

Departamento de Producción

- A. Las capacitaciones a inspectores no se encuentran registradas en documentos, y tratan temas relacionados con las obligaciones y deberes de los funcionarios, precios y horarios que se deben de saber, poniendo especial énfasis en la experiencia acumulada por el grupo de inspección en la solución de los problemas cotidianos.
- B. No existe nada que le muestre los horarios de los cartones para los conductores no cobradores (personal que no necesita de ir a las oficinas de recaudación y que usualmente tiene que llamar por teléfono o ir personalmente para ver sus horarios), o los que se encuentran en su día de descanso.
- C. El personal de inspección para asignar los horarios, utiliza un software que hoy en día no involucra el Smart (falta terminar la funcionalidad dentro del Smart).

Área de Recursos Humanos y Administración

- A. Es sabido que en algunos procesos hay **información** que se encuentra duplicada o inclusive triplicada, haciendo perder tiempo en varias ocasiones re ingresando información.
- B. Falta de comunicación de cierta información, como ser de los beneficios de la Tarjeta “Bus”, la cual tiene beneficios cambiantes y actualmente los empleados de CUTCSA no tienen una forma sencilla de acceder a dicha información.

Departamento de Transporte

- A. Las planillas de estadísticas se encuentran en una etapa de pasaje de las planillas al Smart, pero aún se utiliza principalmente Excel. Hoy se obtienen las estadísticas del día desde las planillas, las cuales se pasan a otras planillas

para hacer los comparativos. Todo este tipo de pasaje hace que se dedique mucho tiempo.

- B. Mejorar las comunicaciones. Por ejemplo: colocar información de forma digital en terminales de autogestión, lo cual ayudaría en rapidez, y economizaría recursos humanos y otros recursos. Hoy, cada aviso debe llevarse vía camioneta a cada línea. Usualmente hay muy poco tiempo para comunicar.
- C. Poder transmitir lo de las carteleras y coches netamente a cada línea de forma digital.
- D. Debe haber algo para los clientes que no sea vía mamparas de los coches.
- E. Que la intendencia municipal o el ministerio pueda comunicarse de otra forma con el departamento (por ejemplo, vía firma digital).

Área Comercial y de Servicios

- A. Los Servicios y Reparaciones no se coordinan mediante un software, pero sí hay sistemas donde se va alimentando de datos cada vez que se le va prestando servicios a coches. Luego del servicio se realiza la facturación de los repuestos (qué compró el coche) y lo que se le cobró mano de obra, y de esta forma se sabe qué se le hizo y qué no a los coches. Además, se cuenta con otro sistema que desde el momento en que se está haciendo el arreglo dentro del taller se puede ir agregando aclaraciones de lo que se le está haciendo al coche.
- B. Las agendas se están empezando a digitalizar para cada taller en los cuales no tenían computadoras hasta diciembre del año pasado (sí había en oficinas de administración). No hay fichas anteriores (viejas), solo facturas de reparaciones que se están pasando a la versión digital.
- C. Cada dueño es responsable de su coche, y debe agendar mantenimiento. La empresa no le avisa a los propietarios sobre los servicios que debe realizar.
- D. Se está pensando hacer un taller mecánico preventivo (hoy lo que más hay es correctivo).
- E. En las unidades nuevas, al ser entregada a sus respectivos dueños, se le da una planilla que indica los kilómetros y los tiempos de mantenimientos que se deben hacer.
- F. Para el caso de repuestos, sí cuentan con un software que tiene estado de las ventas de mostrador y pedidos, el stock mínimo y crítico el cual el software avisa cuando se llegan a los niveles mínimos o críticos de stock, y ofrece un análisis de uso y últimas adquisiciones. Igual, se analiza y se chequea esa advertencia de forma manual para ver si realmente es necesaria la adquisición.

- G. Existen procedimientos o procesos de trabajos que se encuentran por escrito y que ya pasaron por directorio. A su vez, las experiencias de los empleados se transmiten a los funcionarios en reuniones que se realizan. Existe a su vez un proyecto sobre la formación de nuevas generaciones que involucra estrategias de la empresa para los próximos 3 años.
- H. Hay una aplicación móvil para los funcionarios de la empresa (propietarios y funcionarios del coche) para encontrar el coche estacionado en el predio. A la entrada del predio también hay una pantalla que permite mostrar la posición del coche.
- I. Se está desarrollando la extensión del software que hoy permite el ingreso de fichas, y que incluiría tarjetas de visitante, registro de la matrícula de los autos asociado a los funcionarios para llevar registro y control de los autos que se encuentran estacionados dentro de los predios, etc.:-
- J. También se está pensando en el desarrollo de una aplicación para ver los códigos de fallas que salen en el tablero de los coches, los cuales hacen referencia al problema y solución dentro del manual.

12.1.1.2.3. Sistemas actualmente en desarrollo

- A. Se está terminando de desarrollar un sistema de análisis de los vales especiales (no quincenales, sino ocasionales que vienen con autorización y que tienen que ver con haberes o cuidados de coches) generales de CUTCSA. El sistema existía previamente, pero se pasó por un proceso de actualización, mejora e integración al Smart. Toda la información ya existe en la base de datos, falta consumirla y analizarla.
- B. Se comenzará a desarrollar un sistema para los actos electorales y asambleas (cambiar los formatos de los sistemas actuales). El sistema tiene que ver con los actos y el recuento de votos.
- C. Otro sistema que se había comenzado, pero que tuvo que suspenderse para desarrollar sistemas más importantes, es el del "0 accidente" (se premia a choferes por no tener multas o choques durante el año, cuyos premios van desde autos hasta electrodomésticos entre otros). El sistema precisa realizarse reingeniería, y permitiría analizar denuncias, infracciones e inspecciones, y realizar notificaciones o reclamaciones que llevan a que el chofer se encuentre o no dentro del beneficio.
- D. Se encuentra pendiente un sistema para consultar la fecha de pago que CUTCSA tiene con otras empresas (por ejemplo, ANCAP) para que ellos tengan acceso y vean las fechas y facturas que tienen que cobrar.
- E. El sistema de comunicados interno ya se encuentra bastante adelantado (75%-80% realizado). El mismo consiste en una especie de Correo interno

con roles, donde se pasa por un proceso de generación de un documento, revisión, envío, etc. En un principio estaría involucrando sectores donde se realizan un mayor volumen de comunicados: el directorio y gerencias. Más adelante se irán integrando otras áreas.

- F. Otro sistema propuesto tenía que ver con los faltantes y sobrantes: mejora en la gestión de envío hacia el departamento de operaciones para luego hacer descuentos o créditos correspondientes.
- G. También se pretende realizar el sistema para los *pannes* (auditoría), para saber si la suspensión de un servicio es o no justificada, y así determinar si hay que pagarle la hora al personal que estaba realizando el servicio, o aplicarle una suspensión. El mismo se encuentra en fase de prueba por parte de los usuarios finales.

12.1.2. Visión del Producto

12.1.2.1. Objetivo general

Crear una aplicación móvil que ayude y apoye la gestión, comunicación y trabajo que realizan los distintos funcionarios en su día a día, en todas las áreas de la empresa de CUTCSA.

La misma permitirá dar soporte, agilizar y mejorar la organización de las tareas, aumentar la eficiencia y eficacia del funcionario y mejorar o complementar las comunicaciones entre empleados, departamentos y áreas.

El funcionario podrá tener de forma más accesible, rápida y sencilla la información que utiliza durante su jornada laboral, aprovechando las ventajas que ofrecen las funcionalidades de un dispositivo móvil.

12.1.2.2. Requerimientos o propuestas

12.1.2.2.1. Todas las Áreas

- A. Sincronizar el servicio de correo de la empresa en la aplicación móvil y así incentivar y facilitar la comunicación a través de este medio. Incluiría una mensajería interna.
- B. Ver información sobre personal (cargo y función, tareas que realiza, horarios de trabajo, mail, ubicación del trabajo, etc.).

- C. Mostrar ayuda sobre qué hacer paso por paso (ej a la hora de un choque, cómo realizar las tarjetas STM o pases libres). Poder ver distintos protocolos de trabajo, que es lo que tiene que hacer el personal, cómo, y qué es lo que NO tiene que hacer.
- D. Mostrar novedades generales de la empresa (aumentos de sueldo, suba de boletos, feriados, jornadas de vacunación, cambios de horario, etc.).
- E. Avisos y recordatorios (aviso que el sueldo ya fue acreditado en la cuenta bancaria del funcionario, aviso que se encuentran disponibles los *tickets* de alimentación para ser retirados, vencimiento de carne de salud, presentarse en cierto lugar - citaciones, cambios de aceite, gomas, etc.).
- F. Pedir reposición de insumos o reparaciones (rollos de boletos, papeles de impresión, reparación de computadora, reparación de fotocopidora, etc.).
- G. Consultas y estadísticas de valores diarios, mensuales o anuales.

12.1.2.2.2. Departamento de Operaciones

- A. Poder ver información en una aplicación móvil sobre recibos de sueldos (que hoy solo se pueden imprimir en los sistemas de autogestión) o cuenta corriente (que hoy en día se envía por mail o es impreso mensualmente a cada propietario).

12.1.2.2.3. Departamento de Recaudación

- A. Ayudar a la comunicación dentro de la oficina y hacia otras oficinas y sincronización de la información.
- B. En tareas que tiene que ver con mandar un montón de planillas poder digitalizar el proceso.
- C. Que los conductores y guardas puedan ver sobrantes/excedentes de recaudación no solo en las terminales de autogestión, sino que dicha información pueda ser consultada también en la aplicación móvil.

12.1.2.2.4. Departamento de Producción

- A. Poder controlar las unidades en tiempo real mediante aplicación móvil, hoy en día se usa el "*iBus*".
- B. Mostrar los horarios de los cartones para los conductores no cobradores (o sea, personal que no necesita ir a las oficinas de recaudación, y que

usualmente llama o tiene que ir sólo para ver sus horarios) o los que se encuentran en su día de descanso.

- C. Cuando se realiza un cambio o interrupción en un servicio que no va a salir o va a ser reforzado, el mismo sea actualizado para que los horarios “teóricos” reflejen estos pequeños cambios que se sabe se van a dar (o al menos pueda ser advertido al funcionario o usuario).
- D. Mostrar información sobre Servicios de coches (horarios asignados para un coche en cierto día/semana/mes; quién y horario que trabaja, qué coches están trabajando, etc.).
- E. Función para pedido de auxilio de remolques para el funcionario.
- F. Función de botón de pánico.
- G. Ingresar y ver interrupciones al servicio.
- H. Realizar ingresos de *pannes* (cuando un coche se rompió).

12.1.2.2.5. Área de Recursos Humanos y Administración

- A. Poder realizar citaciones para declarar a un empleado mediante la aplicación móvil, al menos de forma alternativa sumando a un envío de mail, mensaje de texto y/o llamado telefónico.
- B. Que los empleados puedan solicitar un vale de sueldo mediante la aplicación móvil.
- C. Que los funcionarios puedan visualizar cuándo es que tienen aprobada una licencia y que se les realice un recordatorio de cuándo tienen que reintegrarse a trabajar.
- D. Mostrar en la aplicación móvil información actualizada y de fácil acceso sobre los beneficios que se pueden usar con la tarjeta “Bus”.

12.1.2.2.6. Departamento de Transporte

- A. Mostrar información a clientes, que hoy se colocan en las mamparas de los coches, dentro de una aplicación.
- B. Centralizar la información de las planillas en un solo lugar, ser consumido desde un solo lugar, y así evitar la duplicación de información.

12.1.2.2.7. Área Comercial y de Servicios

- A. Ver el status de un coche dentro del taller (“En espera”, “Ingresado al taller”, “Pronto para retirar”), y enviar notificaciones al propietario.

- B. Ver agenda del centro de lubricación y alineación, y que pueda ser "autoagendable".
- C. Mostrar información a propietarios sobre reparaciones que se le hizo a su coche, cantidad de combustible que se le cargó, etc.
- D. Avisos por falta de stock de componentes desde la aplicación.
- E. Ver el análisis y estados del stock de componentes desde una aplicación.
- F. Incorporar la aplicación móvil ya existente, que permite encontrar el coche estacionado en el predio de la empresa, a la nueva aplicación.
- G. Poder analizar los códigos de fallas que salen en el tablero de los coches y describir soluciones o referencias al manual.

12.1.2.2.8. Otras propuestas

- A. Que los proveedores del Club CUTCSA puedan acceder a consultas de fechas de pagos, montos, etc.

12.1.3.2. Anexo 2 Proceso de Gestión de Accidente

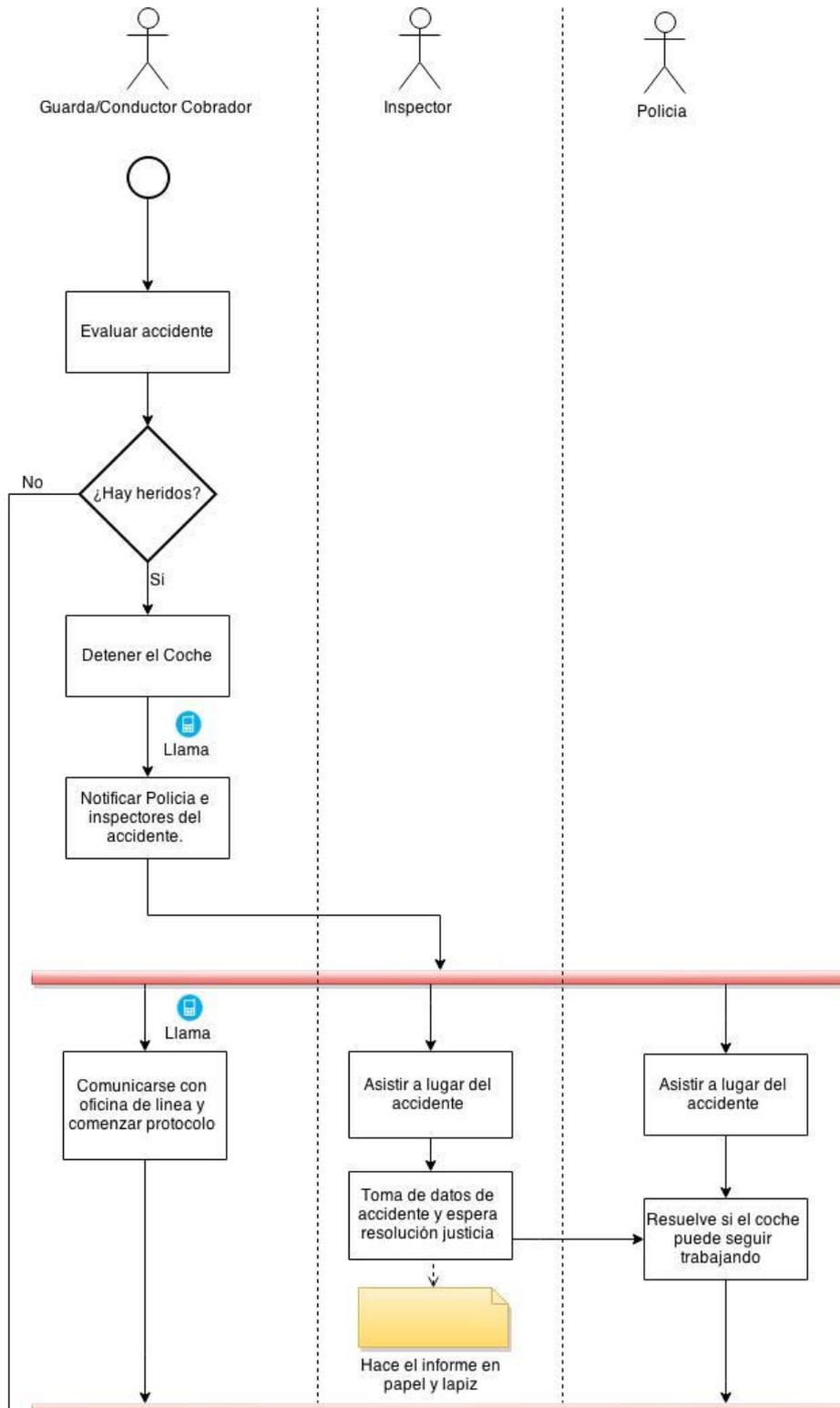


Ilustración 12-2 Proceso de Gestión de Accidente parte 1

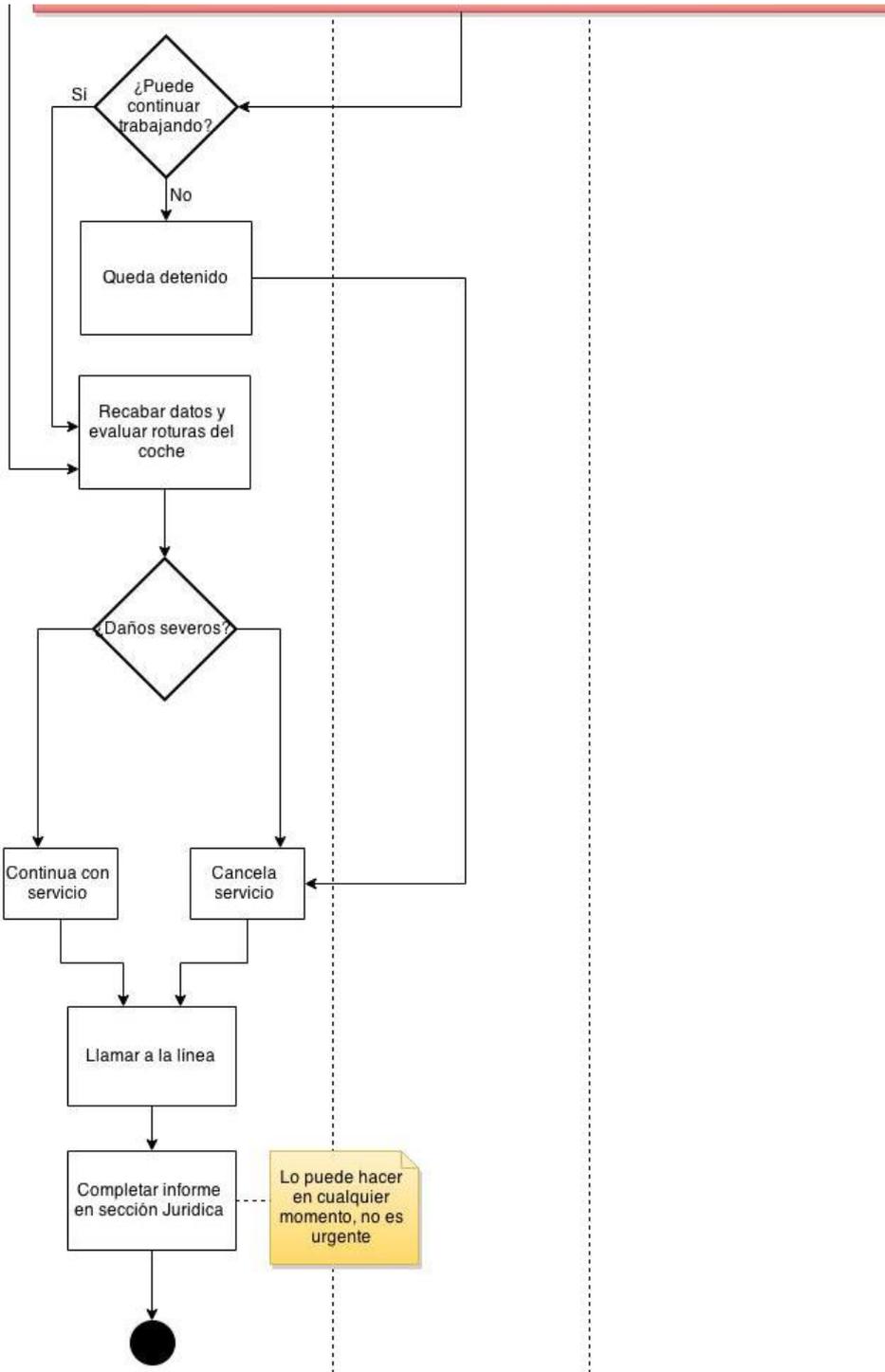


Ilustración 12-3 Proceso de Gestión de Accidente parte 2

12.1.3.3. Anexo 3 Proceso de reparación de un coche

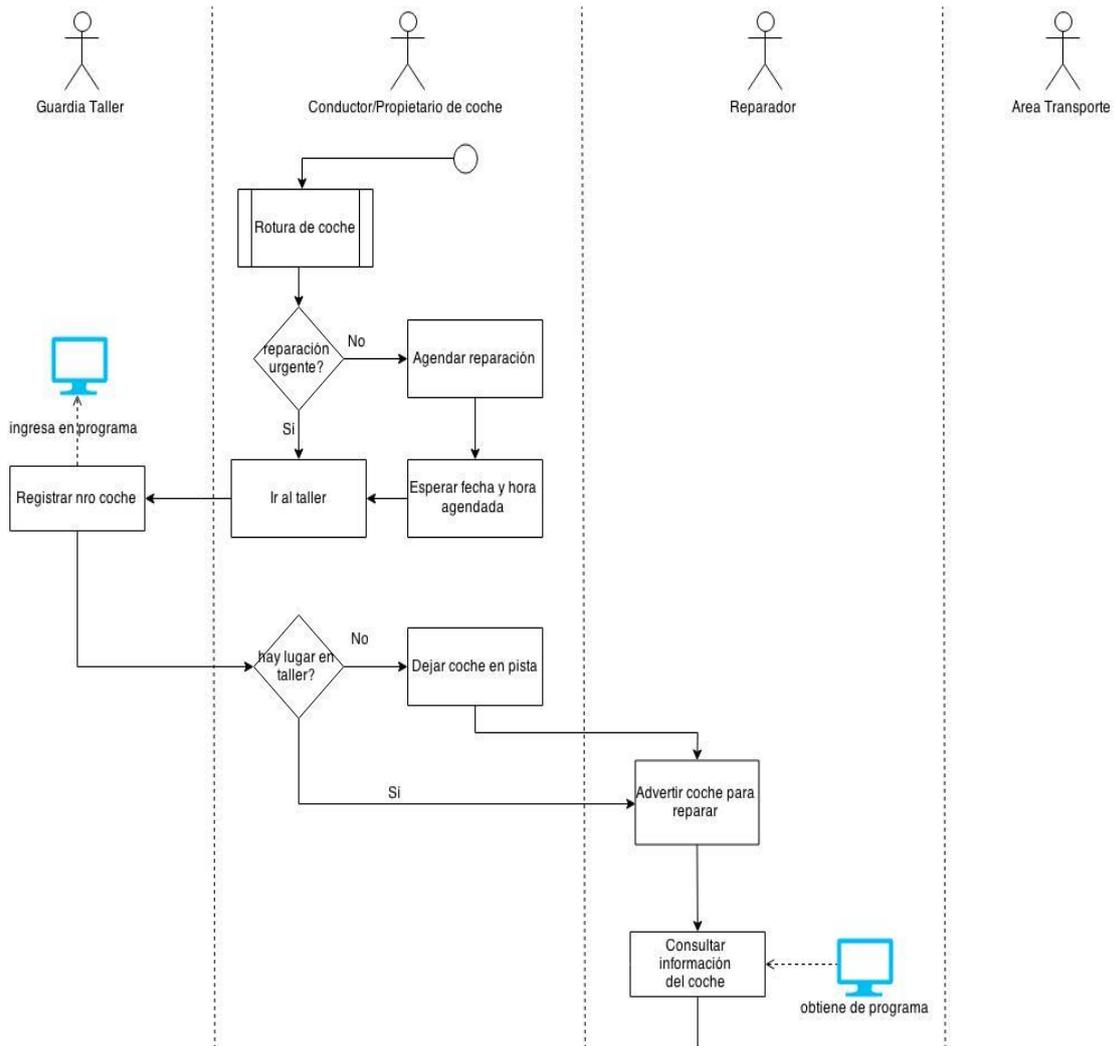


Ilustración 12-4 Proceso de reparación de un coche parte 1

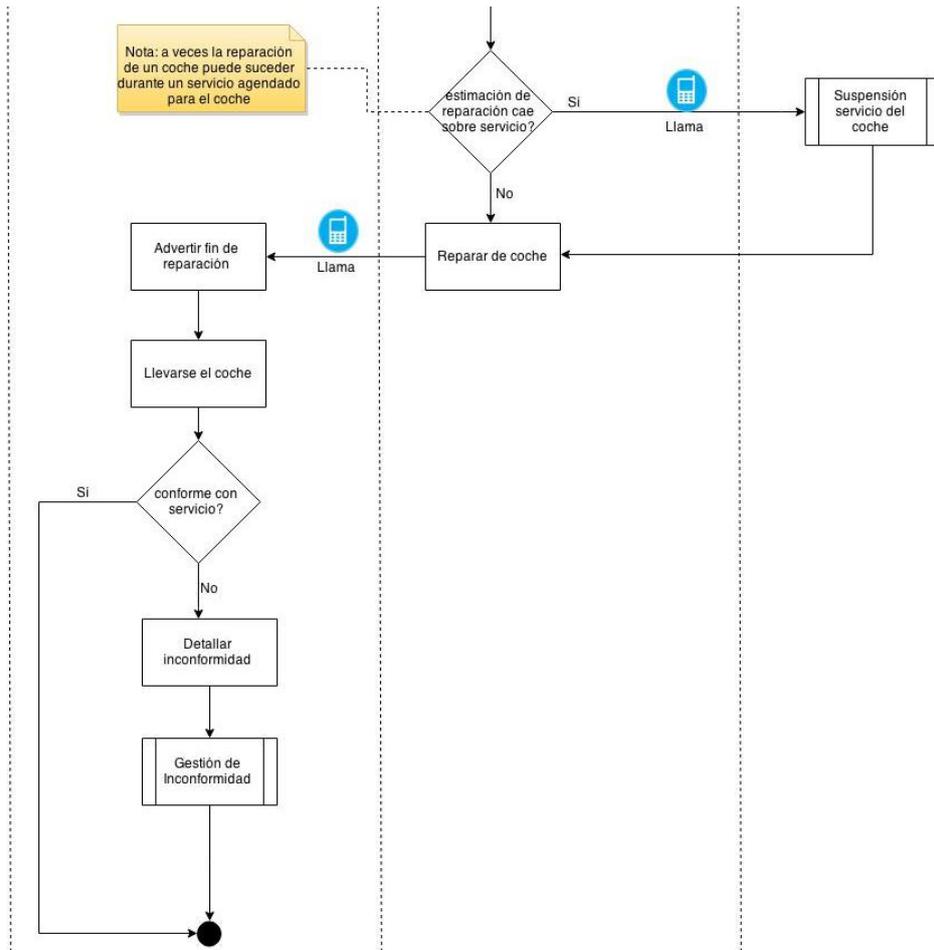


Ilustración 12-5 Proceso de reparación de un coche parte 2

12.1.3.4. Anexo 4 Comunicaciones de la empresa

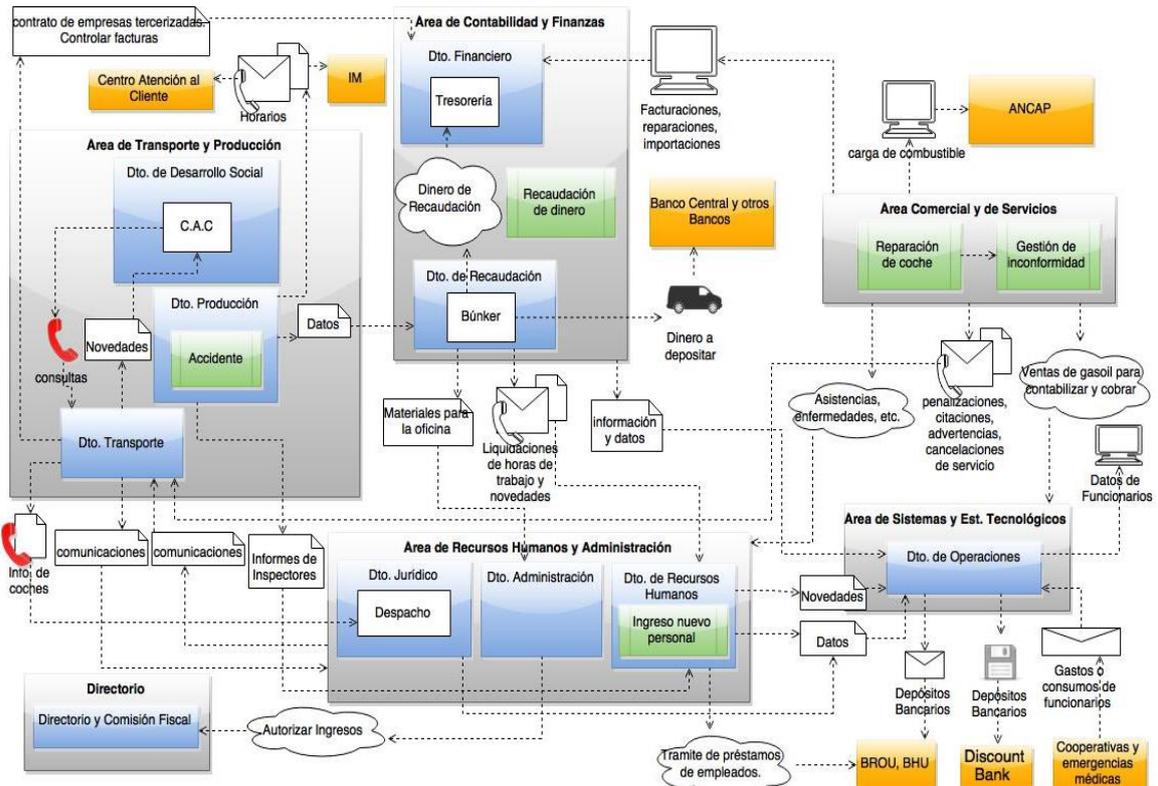


Ilustración 12-6 Diagrama de comunicaciones de la empresa

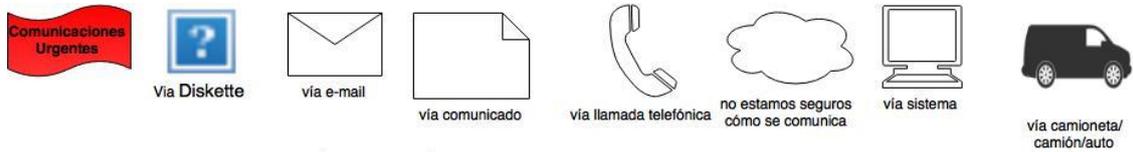


Ilustración 12-7 Referencias al Diagrama de comunicaciones parte 1

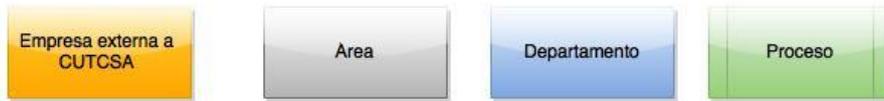


Ilustración 12-8 Referencias al Diagrama de comunicaciones parte 2

12.2. Anexo 2 – ESRE

12.2.1. Introducción

12.2.1.1. Identificación del sistema

El sistema se denominará “Aplicación móvil de apoyo”.

12.2.1.2. Propósito del documento de especificación de requerimientos

El documento de especificación de requerimientos tiene como fin la especificación de los requerimientos funcionales y no funcionales del sistema, además de los distintos casos de uso. El motivo por el cual fue realizado es para que sea comprendido y tenido en cuenta por el equipo de desarrollo del sistema, futuros equipos de desarrollo que se encarguen de mantenerlo o extenderlo y por último para el cliente y usuarios finales del sistema.

12.2.2. Descripción General

12.2.2.1. Perspectiva general del producto

El objetivo del producto es la creación de una aplicación que sirva de apoyo para diferentes áreas de la empresa CUTCSA mediante distintas funcionalidades que esta va a brindar. Va a servir como herramienta de control de algunos procesos, consultas varias de información, recepción de notificaciones y avisos, y de gestión.

12.2.2.2. Funciones principales

El sistema a desarrollar va a tener varias funcionalidades enfocadas a distintas áreas de la empresa. Para el área correspondiente a Transporte y producción, particularmente para los Inspectores. Se listan las funcionalidades a continuación:

- Control de viajes.
- Localizar coches en tiempo real en el mapa.
- Estimación de tiempo de llegada de coche a una parada en base a información de GPS del coche.
- Indicaciones de cómo llegar de un punto a otro en el mapa.
- Ver recorrido de líneas.
- Visualización de horarios teóricos planificados de líneas.

- Mostrar en mapa información de las distintas oficinas de venta de boletos.

Para el área correspondiente a Transporte y producción, particularmente para los Guardas y conductores. Se listan las funcionalidades a continuación:

- Consultar servicio asignado.

Para el área correspondiente a Transporte y producción, particularmente para los largadores. Se listan las funcionalidades a continuación:

- Dar de alta un nuevo servicio.
- Modificación de servicio.
- Eliminar servicio.

Finalmente, todos los usuarios de la aplicación compartirán la siguiente funcionalidad:

- Recepción de avisos y notificaciones.
- Gestión de notificaciones.

12.2.2.3. Características de los usuarios

Vamos a definir los distintos tipos de usuarios que interactuarán con el sistema a desarrollar:

Inspectores

Los inspectores de coches utilizarán la aplicación como herramienta de trabajo en el día a día. La misma les servirá para realizar controles y consultas varias.

Guardas

Los guardas utilizarán la aplicación para obtención de información y notificaciones.

Conductores

Al igual que los Guardas, utilizarán la aplicación para obtención de información y notificaciones.

Conductores Cobradores

Al igual que los Guardas, utilizarán la aplicación para obtención de información y notificaciones.

Largadores

Los largadores podrán hacer altas, modificaciones y eliminar viajes puntuales y además poder recibir avisos y notificaciones.

12.2.3. Requerimientos

12.2.3.1. Requerimientos funcionales

12.2.3.1.1. Requerimientos comunes a todo el sistema

Los requerimientos comunes a todo el sistema son requerimientos que pertenecen y funcionan para todos los usuarios de la aplicación de forma indiscriminada.

Se utilizó el prefijo "G" para identificar los requerimientos comunes a todo el sistema.

[G-RF1: Log In](#)

Descripción: Se deberá de haber iniciado sesión para poder utilizar cualquiera de las funcionalidades de la aplicación. Según el rol del usuario es las funcionalidades que se habilitaran para usar.

[G-RF2: Log Out](#)

Descripción: Al terminar de utilizar la aplicación los usuarios deberán de cerrar su sesión.

[G-RF3: Mostrar Menú](#)

Descripción: Se deberá de poder acceder a un menú donde se tendrá acceso a todas las funcionalidades disponibles para el rol de usuario que inició sesión en la aplicación.

[G-RF4: Reportar un error en la aplicación](#)

Descripción: Se deberá de poder reportar errores en la aplicación desde cualquier sección de la misma que se encuentre navegando.

[G-RF5: Recepción de notificaciones](#)

Descripción: Recepción de notificaciones varias dentro de la aplicación, enviadas desde distintos sectores de la empresa y con destino a usuarios y/o roles específicos. Algunos ejemplos: aviso de asignación de servicio, vencimientos de carné de salud, vencimiento de libreta de conducir, nuevos desvíos planificados, entre otros.

[G-RF6: Gestión de notificaciones](#)

Descripción: Se deberá poder visualizar todas las notificaciones recibidas, viéndolas de manera diferenciada si ya fueron leídas previamente o no, acceder a las mismas, y eliminar las que no se deseen conservar más en la aplicación.

[G-RF7: Visualización de noticias](#)

Descripción: Recepción de noticias referentes a la empresa, las cuales se desplegarán por defecto al iniciar la aplicación, pudiéndose acceder a las mismas luego mediante una opción dedicada en el menú de usuario.

12.2.3.1.2. Requerimientos Módulo Inspectores

Se utilizó el prefijo “C2” para identificar los requerimientos específicos de este módulo.

[C2-RF1: Ver información de paradas](#)

Descripción: Ver las paradas más cercanas a la ubicación del usuario y mostrar información sobre las mismas (número, dirección y líneas que pasan por ella).

[C2-RF2: Visualizar el recorrido de cierta línea en un mapa](#)

Descripción: Mostrar en un mapa el recorrido que tiene cierta línea, indicando en el mapa las paradas por las que pasa.

[C2-RF3: Mostrar arribo de líneas a la parada](#)

Descripción: Mostrar los próximos arribos de líneas a una parada (pudiendo filtrar las búsquedas por parada), para cada servicio próximo a arribar, se podrá ver los siguientes datos: tiempo estimado de llegada a la parada (utilizando GPS a bordo del coche) y ubicación en el mapa.

[C2-RF4: Ubicar coche en un mapa](#)

Descripción: Ver la ubicación e información (número, recorrido, destino) de cierto coche en un mapa (utilizando la información provista por el GPS a bordo del coche).

[C2-RF5: Ver ruta entre un origen y un destino](#)

Descripción: Conocer la ruta y qué ómnibus (solo de la empresa CUTCSA) tomar desde un origen a cierto destino. Se indicará a dónde se deberá caminar y a qué parada dirigirse.

[C2-RF6: Ver información de puntos de interés](#)

Descripción: Ver información (nombre, dirección, teléfono y horarios) de las oficinas de venta de boletos, oficinas de línea y demás dependencias de CUTCSA.

[C2-RF7: Control del servicio](#)

Descripción: Obtener el reporte que en la actualidad se obtiene impreso a través de la expendedora de boletos de un coche, conteniendo la información necesaria para la inspección del servicio. A su vez, poder realizar el control de boletos acercando la tarjeta del pasajero al celular y que el mismo alerte en caso de que la tarjeta no se haya utilizado en el viaje. En caso de que el pasajero no haya adquirido su boleto utilizando tarjeta, se podrá controlar mediante la numeración del boleto. Finalizado el control, quedara guardado un registro del mismo.

[C2-RF8: Mostrar horario planificado de una línea](#)

Descripción: Ver el horario planificado de cada recorrido para cierta línea.

12.2.3.1.3. Requerimientos Módulo Guardas/Conductores/ Conductores cobrados

Se utilizó el prefijo “C3” para identificar los requerimientos específicos de este módulo.

[C3-RF1: Consultar servicios asignados](#)

Descripción: Consultar todos los servicios asignados al usuario para las próximas jornadas laborales.

12.2.3.1.4. Requerimientos Módulo Largadores

Se utilizó el prefijo “C4” para identificar los requerimientos específicos de este módulo.

[C4-RF1: Modificar servicios](#)

Descripción: Poder modificar el horario de salida respecto al planificado y el coche asignado a un servicio.

[C4-RF2: Alta servicio](#)

Descripción: Poder dar de alta un servicio para una fecha y horario determinada.

[C4-RF3: Baja servicio](#)

Descripción: Poder dar de baja un servicio para una fecha y horario determinado.

12.2.3.2. Características de calidad

Se describen a continuación las características de calidad que se espera que el sistema posea, dados los distintos tipos de usuario y los objetivos generales del producto. Para su definición el equipo se basó en la norma vigente a la fecha referente a requerimientos de calidad: ISO/IEC 25010:2011.

RNF01: Fiabilidad – Disponibilidad: El sistema deberá de tener una disponibilidad mínima del 98%.

RNF02: Usabilidad – Facilidad de Operación: Debido a que esta aplicación va a ser una herramienta de trabajo diaria para algunos usuarios, es muy importante que la misma no entorpezca la realización del mismo. El sistema deberá de poder ser controlado con facilidad, por lo que se determinó que todas las operaciones puedan ser accedidas mediante 3 pasos (*clicks*) como máximo.

RNF03: Usabilidad – Estética de la interfaz de usuario: La interfaz de usuario de la aplicación deberá de ser atractiva e intuitiva al momento de realizar cualquiera de las operaciones disponibles. Por lo que se deberán de respetar las pautas de diseño sugeridas por Android.

RNF04: Usabilidad – Facilidad de Aprendizaje: Se plantea un tiempo máximo de 30 minutos para que los usuarios puedan comprender y ser capaces de utilizar las funcionalidades principales sin mayores inconvenientes.

RNF05: Eficiencia – Comportamiento temporal: En condiciones normales de uso del sistema, el mismo no debería de tener una demora mayor a 5 segundos para realizar la operación solicitada por el usuario.

RNF06: Eficiencia – Capacidad: La aplicación deberá de ser capaz de soportar a 5000 usuarios. Se llegó a este número sumando el total de empleados de la empresa. Si bien no se van a alcanzar a todos estos durante el proyecto, los planes de la empresa son extenderlo para todos sus empleados.

RNF07: Mantenibilidad - Modificabilidad: El cliente va a continuar con el desarrollo y extensión de la aplicación luego de culminado el presente proyecto, por lo que es de vital importancia que el sistema pueda ser modificado fácilmente y eficientemente. Es requisito contar con una especificación de casos de uso, diagramas de paquetes, despliegue, clases y componentes.

RNF08: Seguridad - Confidencialidad: Las distintas operaciones ofrecidas por la aplicación solo deberán de ser accedidas por usuarios cuyo rol corresponde con un rol habilitado para acceder a dicha operación.

RNF09: Seguridad – Autenticidad: Se deberá garantizar la identidad de quienes van a realizar cualquier operación ofrecida por el sistema.

RNF10: Seguridad – Integridad: Se deberá de garantizar la prevención del acceso y/o modificación de datos no autorizado a los datos de la aplicación y a la aplicación en si misma.

12.2.3.3. Restricciones

RNF11: Desarrollo en Android: Descripción: El sistema debe ser desarrollado para Android versión 4.0.3 y posteriores.

RNF12: Compatibilidad con Samsung Galaxy ACE 3: El móvil en el que la aplicación va a ser utilizado es de marca: Samsung, modelo: Galaxy ACE 3. Por lo que tendrá que ser totalmente compatible con el mismo, ya sea en términos de funcionamiento y visualización.

RNF13: Desarrollo en .Net con C# para el *backend*: Es necesario que el *backend* de la aplicación esté desarrollado en tecnología .NET y más precisamente utilizando lenguaje C#.

12.2.4. Casos de Uso

Se realizaron bocetos de los distintos casos de uso, los mismos son meramente ilustrativos y en forma de guía. No son la versión final de cómo se verá la aplicación.

12.2.4.1. Casos de uso comunes a todo el sistema

Se utilizó el prefijo “G” para identificar los requerimientos comunes a todo el sistema.

12.2.4.1.1. G-CU1–Log In

RF asociado	G-RF1
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	1. El dispositivo debe tener acceso a internet.

Tabla 12-1 G-CU1

Curso de la funcionalidad (con interfaz de ejemplo):

1. Ejecutar la aplicación en el dispositivo móvil.

2. Introducir los datos pedidos (usuario y contraseña) e iniciar sesión en el sistema.

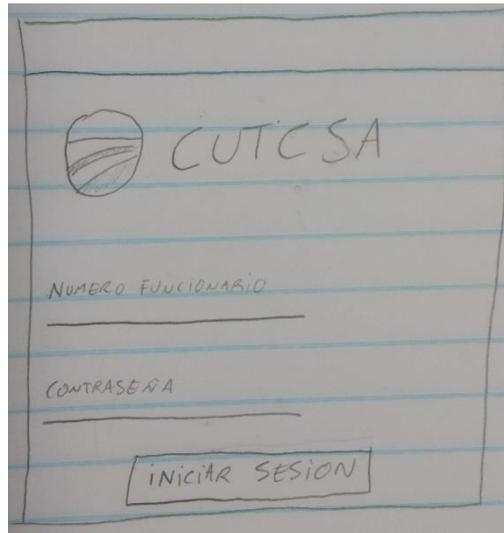


Ilustración 12-9 ventana login

3. Se accede a la aplicación y se muestran solo las funcionalidades específicas para el rol que tiene iniciada la sesión.

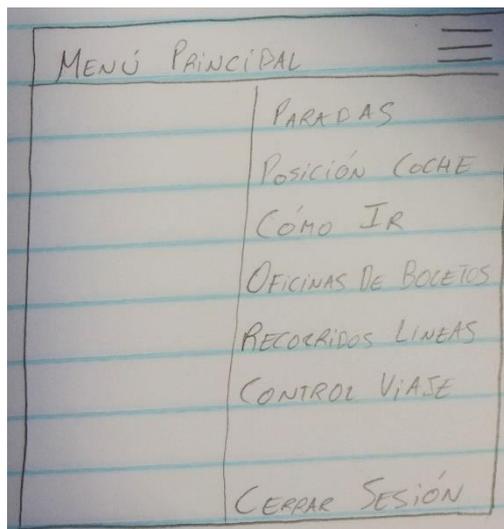


Ilustración 12-10 menú principal

- 3.1 En caso de que el número de funcionario o contraseña ingresados sean incorrectos se desplegara el siguiente mensaje "Número de funcionario o contraseña incorrectos, verifique" y se volverá a paso número 2.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: "Error de Conexión".

12.2.4.1.2. G-CU2–Log Out

RF asociado	G-RF2
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	No Hay.

Tabla 12-2 G-CU2

Curso de la funcionalidad (con interfaz de ejemplo):

1. Se selecciona la opción de cierre de sesión de la aplicación.

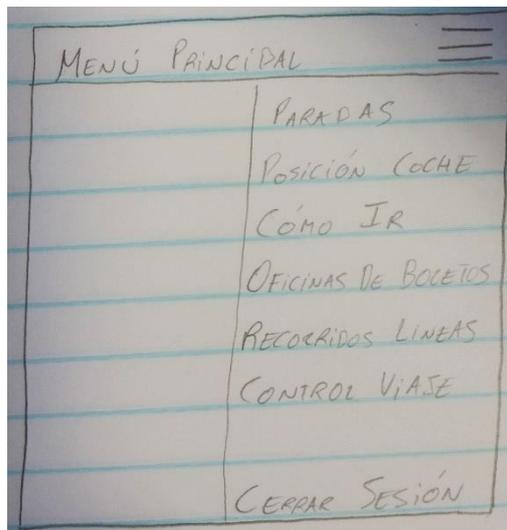


Ilustración 12-11 menú principal

2. Se cierra la sesión y se vuelve al menú del Inicio de Sesión, ver Caso de Uso: [G-CU1 – Login](#)

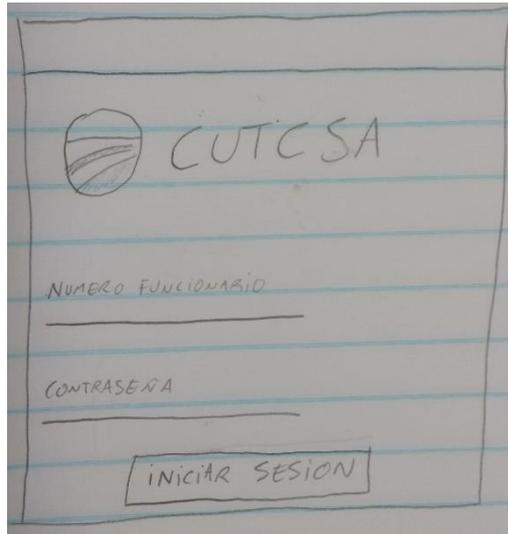


Ilustración 12-12 login

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”.

12.2.4.1.3. G-CU3 – Mostrar Menú

RF asociado	G-RF3
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	1. usuario logueado a la aplicación.

Tabla 12-3 G-CU3

Curso de la funcionalidad (con interfaz de ejemplo):

1. Se selecciona la opción de “Menú” en la aplicación móvil.
2. Se despliegan todas las opciones que brinda el menú para el rol que se encuentra logueado en el sistema.

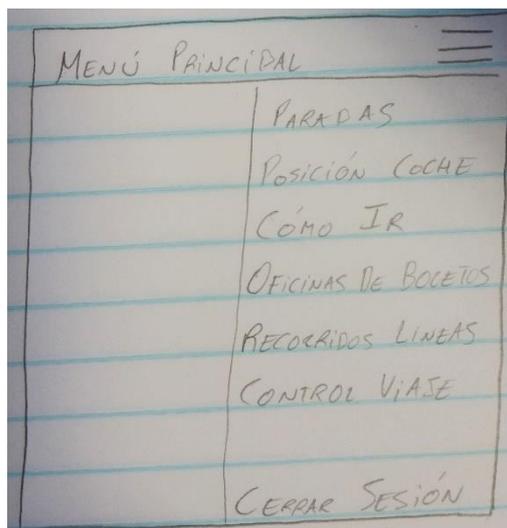


Ilustración 12-13 menú principal

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Opciones de menú incorrectas”.

12.2.4.1.4. G-CU4 – Reportar error

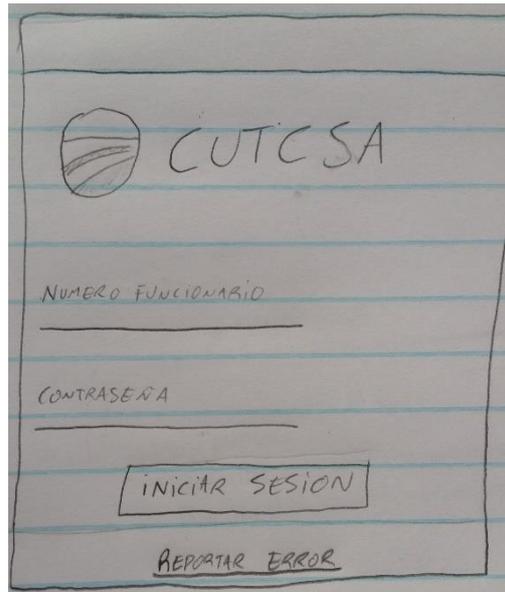
RF asociado	G-RF4
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	Estar ejecutando algún CU

Tabla 12-4 G-CU4

Curso de la funcionalidad (con interfaz de ejemplo):

1. Durante todo momento en la ejecución de cualquiera de los CU del sistema, se puede acceder al presente CU mediante la opción “Reportar

Error”.



2.

Ilustración 12-14 login

3. Se despliegan en pantalla la lista de posibles errores a reportar, los cuales varían según desde qué CU se accede, en cada uno se explicitan.

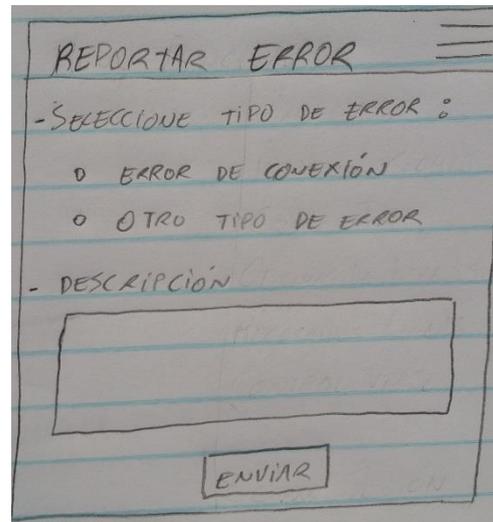


Ilustración 12-15 reportar error

4. El usuario selecciona un error de la lista o en su defecto la opción “otro”.
5. Se escribe una descripción de lo ocurrido y se confirma, enviándose el reporte a los desarrolladores.
 - 4.1 Si se cancela el reporte se vuelve al CU desde cual se invocó (que se estaba ejecutando previamente).

12.2.4.1.5. G-CU5 – Recepción de notificaciones

RF asociado	G-RF5
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1 Usuario logueado a la aplicación. 2 El dispositivo debe de tener acceso a internet 3 Se de ingresar al menos una nueva notificación

Tabla 12-5 G-CU5

Curso de la funcionalidad (con interfaz de ejemplo):

1. Indiferentemente si la aplicación móvil está activa o en plano secundario.
2. Al recibir una nueva notificación se despliega un nuevo aviso en la barra de notificaciones de Android alertando al usuario de la misma.

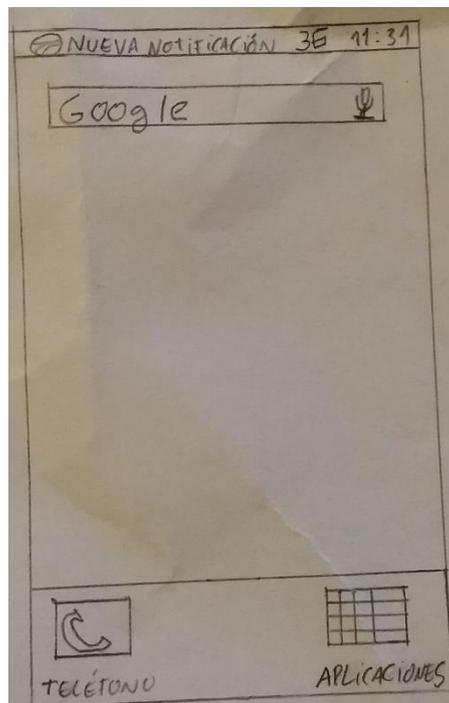


Ilustración 12-16 nueva notificación

3. Cliqueando sobre la misma, la aplicación accede al menú de notificaciones de la aplicación y desplegando la nueva notificación recibida.

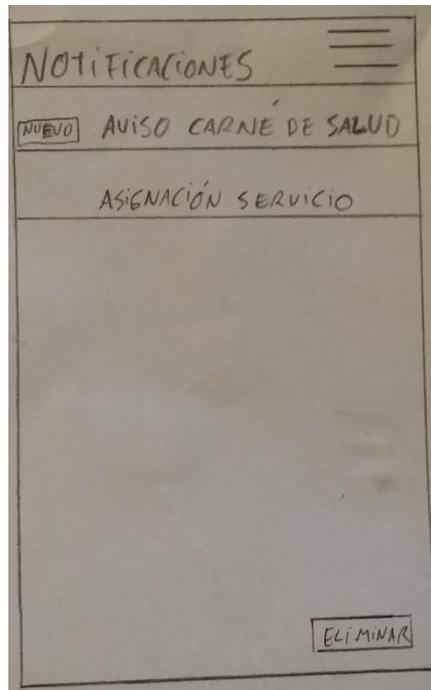


Ilustración 12-17 notificaciones

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Notificación incorrecta”.

12.2.4.1.6. G-CU6 – Gestión de notificaciones

RF asociado	G-RF6
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1 Usuario logueado a la aplicación. 2 El dispositivo debe de tener acceso a internet

Tabla 12-6 G-CU6

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad dentro del menú.
2. Se despliegan las notificaciones recibidas por el usuario, destacando las no leídas hasta el momento.

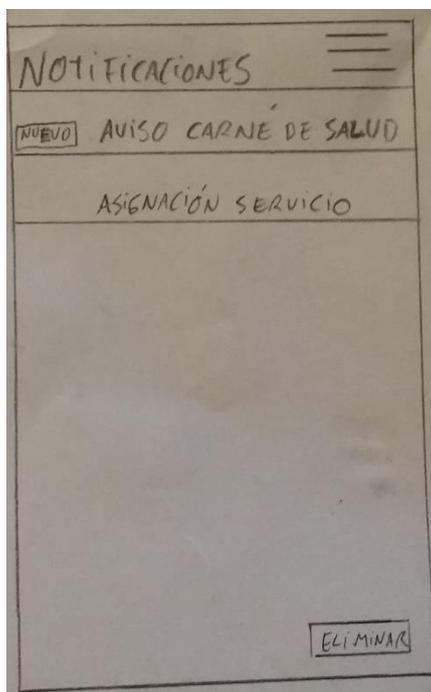


Ilustración 12-18 notificaciones

- 2.1 En caso de no tener ninguna notificación, la lista estara vacia. Fin del CU.
3. Para poder ver el detalle completo de una notificación, se debera acceder a la misma y luego se desplegara en pantalla.
4. Se podra eliminar notificaciones desde la lista principal de notificaciones, seleccionandolas, confirmando su borrado, y finalmente se actualizara la lista de notificaciones, quitando la/s borrada/s.

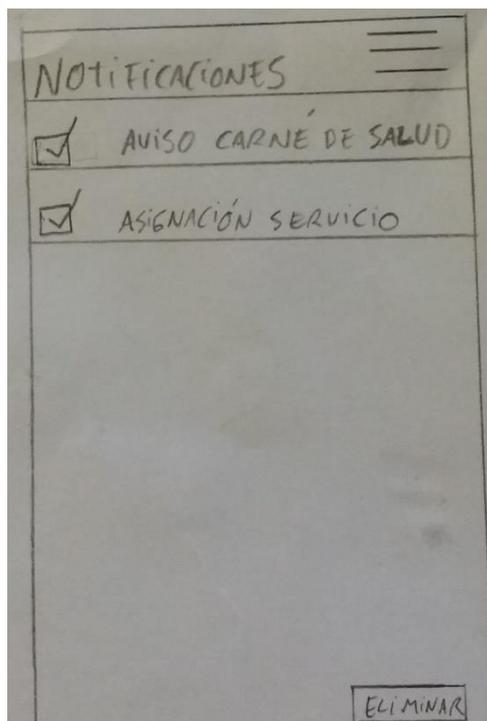


Ilustración 12-19 notificaciones

4.1 En caso de haber eliminado todas las notificaciones, se pasa al punto 2.1 de este CU.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Notificación incorrecta”, “No se eliminaron notificaciones”.

12.2.4.1.7. G-CU7 – Visualización de noticias

RF asociado	G-RF7
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector o Guarda o Conductor o Conductor-Cobrador o Largador.
Permisos para acceso a datos	Público
Precondición	1 El dispositivo debe de tener acceso a internet

Tabla 12-7 G-CU7

Curso de la funcionalidad (con interfaz de ejemplo):

1. El usuario se loguea en el sistema.
2. Se despliegan las noticias nuevas en pantalla.

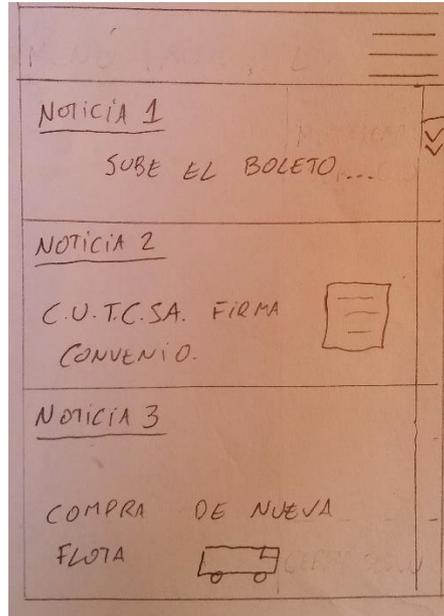


Ilustración 12-20 noticias

2.1 En caso de no tener ninguna noticia, la lista estara vacia. Fin del CU.

3. Para poder ver el detalle completo de una noticia, se debera acceder a la misma y luego se desplegara en pantalla.

NOTA: Para el caso de acceder al [Caso de Uso G-CU4](#) Reportar Error, se hará con los siguientes posibles errores como parámetros: "Noticia incorrecta.

12.2.4.2. Casos de uso modulo Inspectores

Se utilizó el prefijo "C2" para identificar los casos de uso específicos de este módulo.

12.2.4.2.1. C2-CU1 - Ver información de paradas

RF asociado	C2-RF1
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector

Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet. 3. El dispositivo debe tener GPS encendido.

Tabla 12-8 C2-CU1

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.
2. Mostrar las posiciones de las paradas sobre el mapa. A su vez, permitir que el usuario escriba una dirección y así obtener las paradas más cercanas a dicha ubicación. También poder ingresar el número de parada y ubicarla en el mapa.



Ilustración 12-21 búsqueda paradas

- 3 Al seleccionar una parada, mostrar su número y las líneas que pasan por ella.



Ilustración 12-22 selección de parada

- 3.1 Si el número de parada ingresado no existe se desplegará el mensaje “La parada ingresada no existe” y se volverá al paso 2 del caso de uso.
4. Al seleccionar la función de información, mostrar las distintas líneas y recorridos, y opciones para arribo de ómnibus, horarios planificados y rutas. Permite el filtrado de líneas para ver opciones de una en particular.

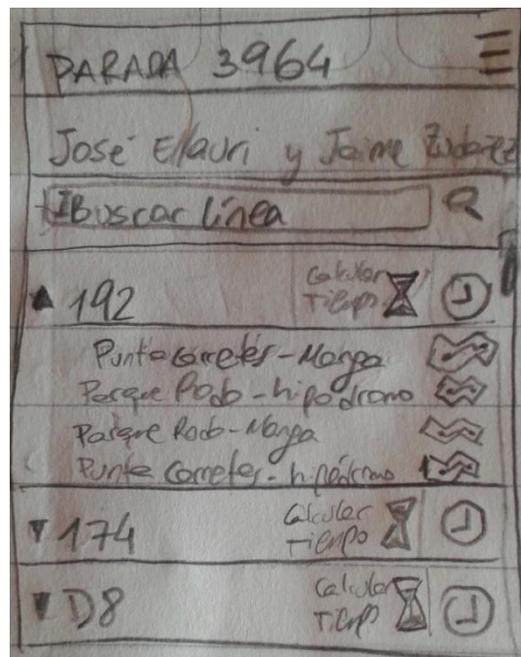


Ilustración 12-23 detalle de parada

- a. Al seleccionar la función de calcular tiempo, ir al caso de uso del [C2-CU3: Mostrar arribo de líneas a la parada](#).
- b. Al seleccionar la función de obtener horario planificado, ir al caso de uso [C2-CU8: Mostrar horario teórico de una línea](#).
- c. Al seleccionar un recorrido particular de cierta línea, ir al caso de uso [C2-CU2: Visualizar el recorrido de cierta línea en un mapa](#).

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Información de parada incorrecta”, “Parada no existente”, “Información de línea incorrecta”, “Línea no existente”.

12.2.4.2.2. C2-CU2 – Visualizar el recorrido de cierta línea en un mapa

RF asociado	C2-RF2
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet.

Tabla 12-9 C2-CU2

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.
2. Permitir que el usuario ingrese el número de línea y recorrido que quiere ver en el mapa. Permite, a su vez, seleccionar sobre una parada y ver su información (Ver Caso de Uso [C2-CU1: Ver información de paradas](#)).

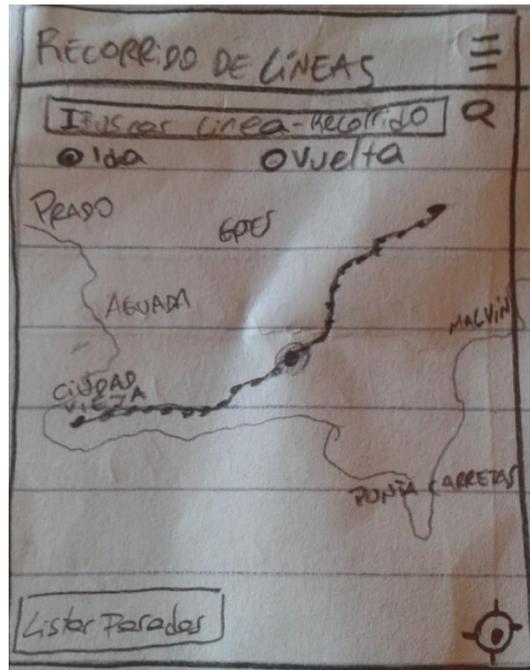


Ilustración 12-24 recorrido líneas

- Al seleccionar la opción para listar las paradas, se muestra la lista de todas las paradas del recorrido. Permite seleccionar una parada y mostrarla en el mapa (Ver Caso de Uso [C2-RF1: Ver información de paradas](#)).

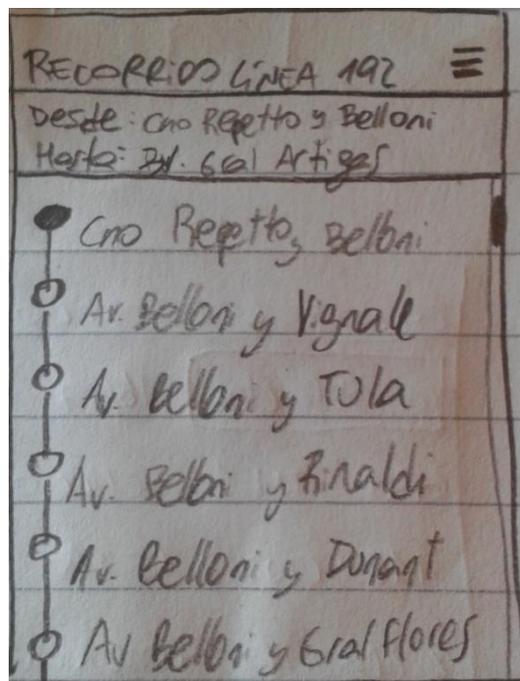


Ilustración 12-25 recorrido línea

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Recorrido incorrecto”, “Recorrido no existente”.

12.2.4.2.3. C2-CU3 – Mostrar arribo de líneas a la parada

RF asociado	C2-RF3
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Público
Precondición	1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet.

Tabla 12-10 C2-CU3

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.
2. Permitir buscar una parada (digitada o mediante la lectura de QR que hay en las paradas) y una línea en particular, y mostrar los próximos arribos de los coches a la parada seleccionada.

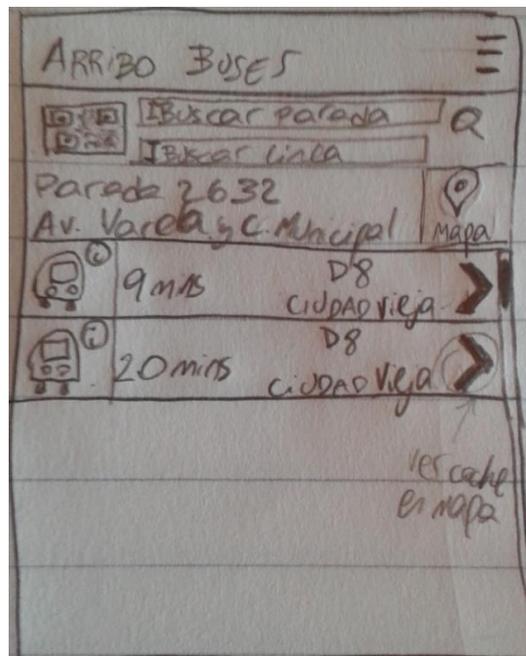


Ilustración 12-26 arribo bus

1. Al seleccionar la opción de ubicar la parada en el mapa, ir al Caso de Uso [C2-CU1: Ver información de paradas.](#)
2. Al seleccionar la opción de ver la información de la línea, ir al Caso de Uso [C2-CU2: Visualizar el recorrido de cierta línea en un mapa.](#)
3. Al seleccionar sobre un pronóstico del arribo, ir al Caso de Uso [C2-CU4: Ubicar coche en un mapa.](#)

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Información de parada incorrecta”, “Tiempo de arribo incorrecto”, “Información de línea incorrecta”, “Línea no existente”.

12.2.4.2.4. C2-CU4 – Ubicar coche en un mapa

RF asociado	C2-RF4
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet.

Tabla 12-11 C2-RU4

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.
2. Permitir buscar un coche en particular, y mostrar su ubicación en el mapa.

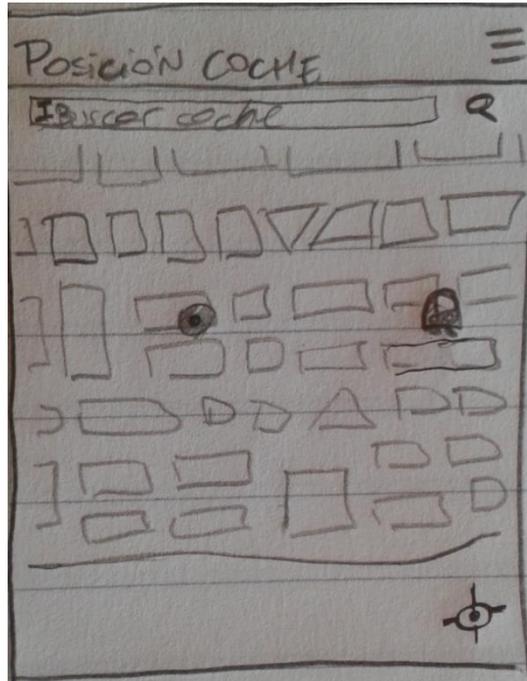


Ilustración 12-27 posición coche

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 ReportarG-CU4 – Reportar error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Ubicación del coche es incorrecta”, “Coche no existente”.

12.2.4.2.5. C2-CU5 – Ver ruta entre un origen y un destino

RF asociado	C2-RF5
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet. 3. El dispositivo debe tener GPS encendido (en caso de utilizar ubicación actual).

Tabla 12-12 C2-CU5

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.

2. Obtener el punto de origen del viaje y mostrarlo en el mapa, se deberá permitir que el usuario escriba una dirección, o también poder marcarlo manualmente en el mapa,



Ilustración 12-28 como ir

- 2.1 Si se escribió la dirección y la misma es incorrecta se desplegará el siguiente mensaje de error “La dirección ingresada es incorrecta” y se volverá al paso 2 del caso de uso.

3. Obtener el punto de destino del viaje y mostrarlo en el mapa, se deberá permitir que el usuario escriba una dirección, o también poder marcarlo manualmente en el mapa,

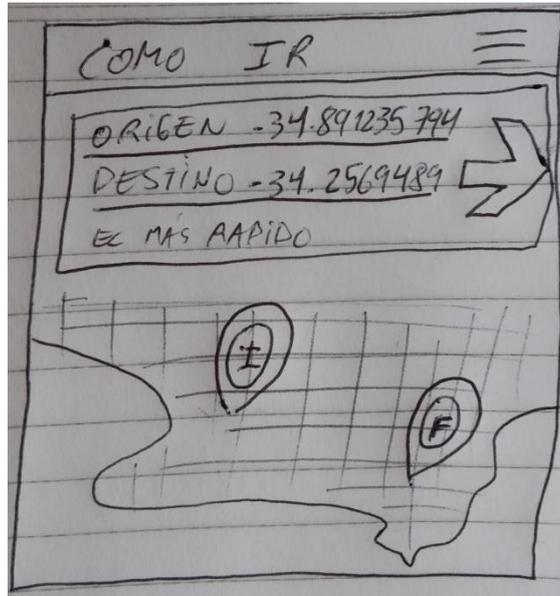


Ilustración 12-29 como ir con coordenadas

- 3.1 Si se escribió la dirección y la misma es incorrecta se desplegará el siguiente mensaje de error “La dirección ingresada es incorrecta” y se volverá al paso 2 del caso de uso.

4. Mostrar en el mapa los distintos pasos para poder ir desde el origen al destino indicados.

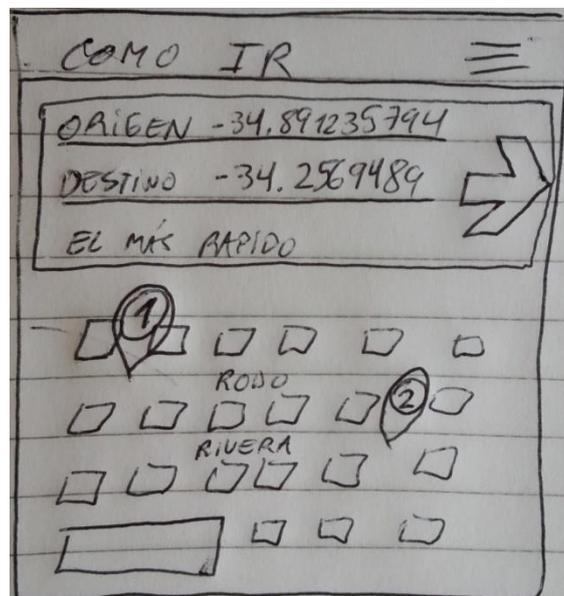


Ilustración 12-30 como ir pasos

4.1 En caso de no existir forma de llegar desde el origen al destino indicado, se desplegará el siguiente mensaje de error “No existe forma de llegar desde el origen al destino indicado” y continúa con el paso 2 del caso de uso.

5. Al seleccionar un paso del viaje se mostrará los detalles e indicaciones del mismo.

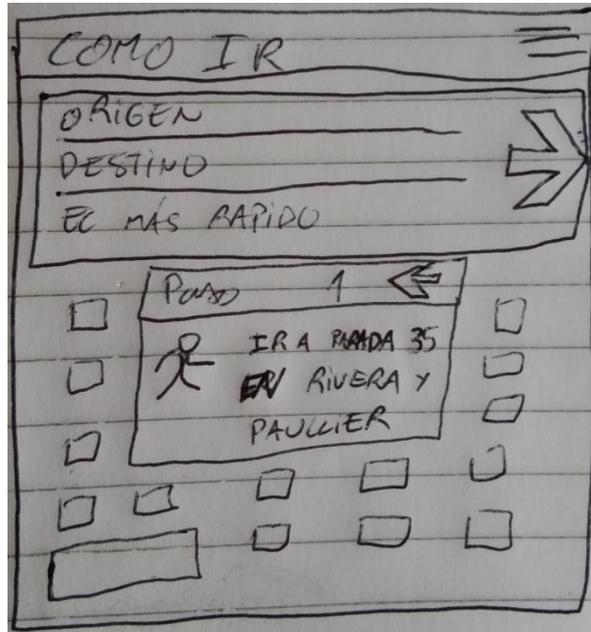


Ilustración 12-31 paso 1

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Información de parada incorrecta”, “Dirección no existe”, “Ruta incorrecta”, “Línea no existente”, “Parada no existente”, “Parada incorrecta”.

12.2.4.2.6. C2-CU6 – Ver información de puntos de interés

RF asociado	C2-RF6
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Público
Precondición	<ol style="list-style-type: none"> 1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet. 3. El dispositivo debe tener GPS encendido (en caso de utilizar ubicación actual).

Tabla 12-13 C2-CU6

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.
2. Mostrar las posiciones de puntos de interés sobre el mapa. A su vez, permitir que el usuario escriba una dirección y así obtener los puntos de interés más cercanas a dicha ubicación. También poder ingresar el nombre de la oficina y ubicarla en el mapa.



Ilustración 12-32 oficina de boletos

3. Al seleccionar un punto de interés, mostrar su nombre, horarios de atención, y teléfono.

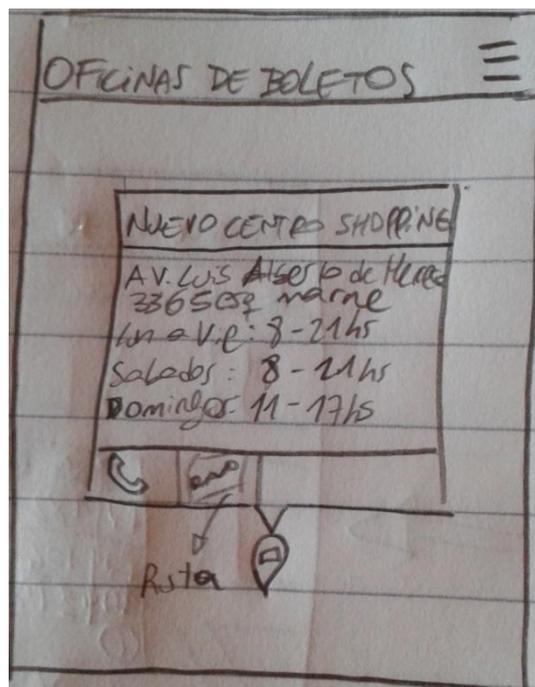


Ilustración 12-33 detalle oficina de boletos

- Al seleccionar la función de ruta, ir al caso de uso del [C2-CU5: Ver ruta entre un origen y un destino](#), con origen la posición actual del usuario, y destino el punto de interés seleccionado.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Información de parada incorrecta”, “Parada no existente”, “Información de línea incorrecta”, “Línea no existente”.

12.2.4.2.7. C2-CU7 – Control de servicio

RF asociado	C2-RF7
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Confidencial (Solo para el inspector logueado)
Precondición	<ol style="list-style-type: none"> Inspector logueado a la aplicación. El dispositivo debe tener acceso a internet. El dispositivo debe contar con lector NFC nxp encendido.

	4. El inspector debe contar con su tarjeta de inspección con datos del servicio pre cargados.
--	---

Tabla 12-14 C2-U7

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.

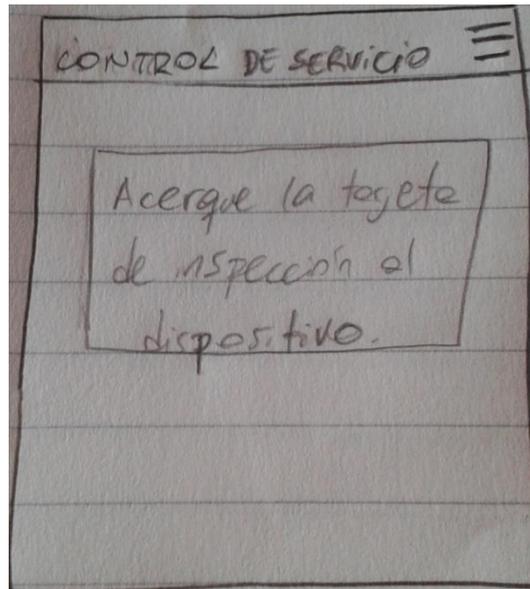


Ilustración 12-34 control servicio tarjeta

2. Aproximar la tarjeta cerca del dispositivo.
 - 2.1 En caso de ser una tarjeta inválida, o que no contiene una inspección pre cargada, se le mostrara el siguiente mensaje “La tarjeta ingresada no es válida” o “La tarjeta no contiene ninguna inspección pre cargada” respectivamente, y se continuará con el paso 2 del caso de uso
3. Se despliega la información necesaria para la inspección del servicio.

CONTROL DE SERVICIO

Control Boletos 16/07 08:56

GR:1 TM:1 Servicio:3

116- CIUDAD VIEJA

Línea:116 conex.125

sale:07:21 llega:08:00

SAM:60096-E04EF3C920B73

Boletos Anulados:

Ilustración 12-35 información control1

CONTROL DE SERVICIO

Boletos Anulados

0000006831	16/07 08:00	\$24
0000006840	16/07 08:05	\$24
0000006845	16/07 08:30	\$24

Horas de pasaje

1	1 Pocitos	07:21
9	1 Pq. el Rob	07:42

Ilustración 12-36 información control 2

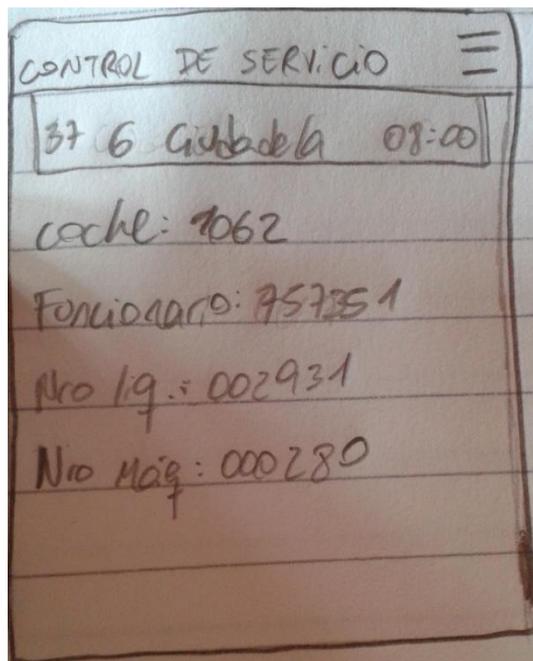


Ilustración 12-37 información control 3

Detalle de datos a mostrar en pantalla:

- Fecha y hora de comenzado el control del servicio
- Número del grupo de rotación (GR)
- Tipo de minuta (TM)
- Número del servicio
- Línea planificada
- Recorrido planificado
- Número de conexión (un número que asocia al viaje)
- Horario planificado de comienzo (salida) y fin (llegada) del servicio
- Primer y último boleto vendido justo antes de obtenerse el reporte de inspección
- Número del SAM que se encuentra dentro de la máquina (“básica”). En caso de no contener ninguna, aparece FFFFF-FFFFFFFFFFFFFFFF
- Descripción de los boletos anulados: número de boleto, fecha, hora y precio
- Horas por la que pasó por cada punto de control: secuencial del punto de control, zona del boleto (un id que representa si es común, céntrico, etc.), el nombre del punto de control, hora planificada que debió pasar por cada punto de control
- Coche
- Número de funcionario del guarda
- Número de liquidación

- Número de máquina (“básica”)

4. Selecciona la opción para iniciar el control de boletos.

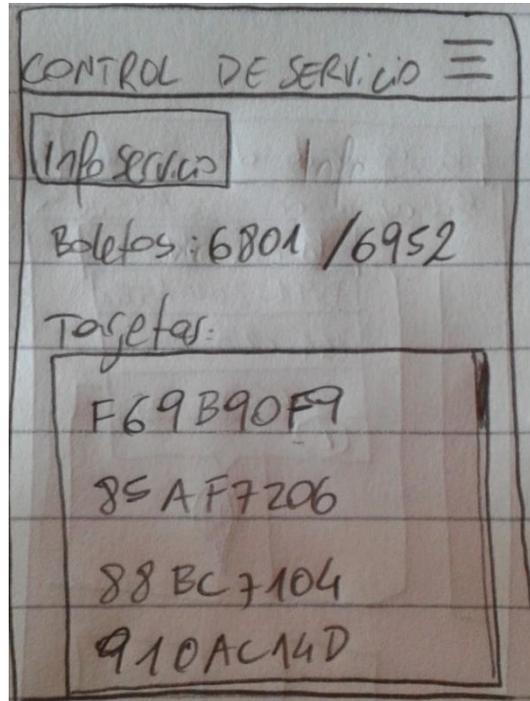


Ilustración 12-38 control

Nota: Si bien no está ilustrado en el boceto, se mostrará en pantalla también la hora en la que se pasó por la expendedora cada tarjeta o que se vendió el boleto.

5. Por cada pasajero, acerca su tarjeta al dispositivo.

5.1 Si la tarjeta aparece en el listado de tarjetas del viaje, se le advierte al Inspector que dicha tarjeta fue usada en el viaje actual con el siguiente mensaje: “Viaje válido”.

5.2 Si la tarjeta no aparece en el listado de tarjetas del viaje, se le muestra un mensaje al Inspector que indica que dicha tarjeta no fue usada en el viaje actual. El mismo dice “Viaje inválido”.

6 Finalizado el control, el sistema guardará el mismo junto con todos sus datos asociados: Nro. coche, Recorrido, Sentido, Hora inicio control, Hora fin de control, todas las tarjetas controladas.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Información del control errónea”, “Problema al leer tarjeta”, “Validación de tarjeta incorrecto”.

12.2.4.2.8. C2-CU8 – Mostrar horario planificado de una línea

RF asociado	C2-RF8
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Inspector
Permisos para acceso a datos	Publico
Precondición	1. Inspector logueado a la aplicación. 2. El dispositivo debe tener acceso a internet.

Tabla 12-15 C2-CU8

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.
2. Permitir seleccionar una línea y la fecha que se quiere obtener la información de los horarios.

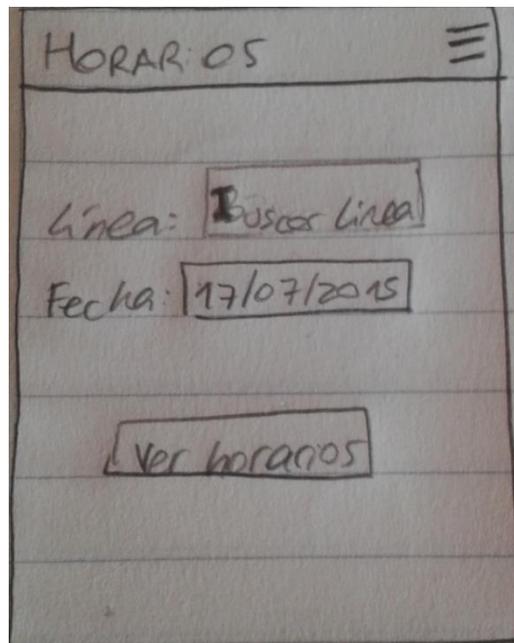


Ilustración 12-39 horarios

- Mostrar todos los recorridos de la línea seleccionada para la fecha seleccionada, pudiendo elegir si el recorrido es de ida o de vuelta.

Horarios 192	
Horario de ida	Horario vuelta
Origen	Destino
Parque Lab 00:04	Manga 00:52
Parque Lab 00:55	Manga 01:47
Parque Lab 02:43	Manga 03:29
Hipódromo 04:26	Manga 04:12
Parque Lab 04:28	Manga 04:37
Hipódromo 04:49	Manga 05:16

Ilustración 12-40 horarios línea

- Al seleccionar un recorrido y horario particular, mostrar el horario que debe pasar por cada parada o puntos de control.

Horarios 192	
< Parque Lab - Manga >	
00:04 - 00:52	
● Salto y P.era	00:04
○ Salto y Ramirez	00:04
○ Ramirez y Poulter	00:05
○ Reising y Girab. -	00:05
○ Reising y Liray	00:06
○ Reising y Figuerie	00:06

Ilustración 12-41 horario recorrido

4.1 Al seleccionar sobre una parada en particular, ir al Caso de Uso [C2-CU1: Ver información de paradas.](#)

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Horario de Línea incorrecto”, “Información de línea incorrecta”, “Línea no existente”, “Recorrido incorrecto”.

12.2.4.3. Casos de uso modulo Guardas y Conductores

12.2.4.3.1. C3-CU1 – Consultar servicios asignados

RF asociado	C3-RF1
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Guarda o Conductor o Conductor-Cobrador
Permisos para acceso a datos	Publico
Precondición	<ol style="list-style-type: none">1. Guarda o Conductor logueado a la aplicación.2. El dispositivo debe tener acceso a internet.3. Tienen que estar cargadas las asignaciones de servicios en el sistema.

Tabla 12-16 C3-CU1

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.

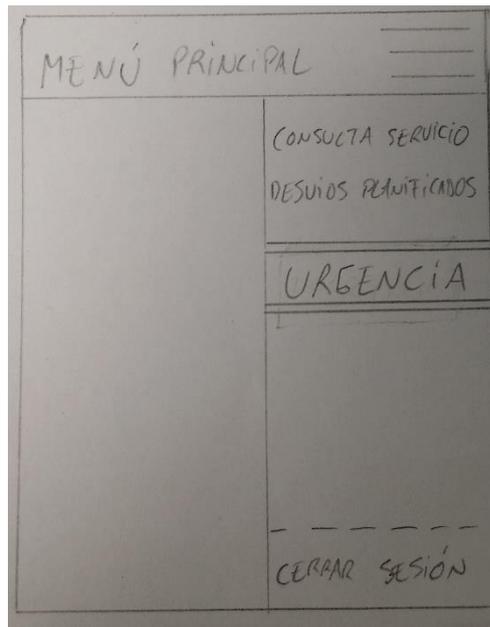


Ilustración 12-42 menú principal

2. Permite visualizar el servicio asignado al usuario para las próximas jornadas laborales (inclusive la del mismo día que se consulta en caso de que la jornada no haya comenzado)

CONSULTA SERVICIOS	
NOMBRE FUNC.	FECHA
MARIO MARTINEZ	19/12/15
EMICIO YANZ	19/12/15
DIEGO CORTINA	19/12/15

3.

Ilustración 12-43 consultar servicios

- 2.1 En caso de no tener servicio asignado para las próximas jornadas, se notificará al usuario con un mensaje que diga "No tiene ningún servicio asignado"

4. Al seleccionar un servicio, se mostrará mayor detalle de el mismo: número de coche, nombre del personal asignado, cargo del personal asignado, número interno del personal asignado, horario, turno, lugar relevo.

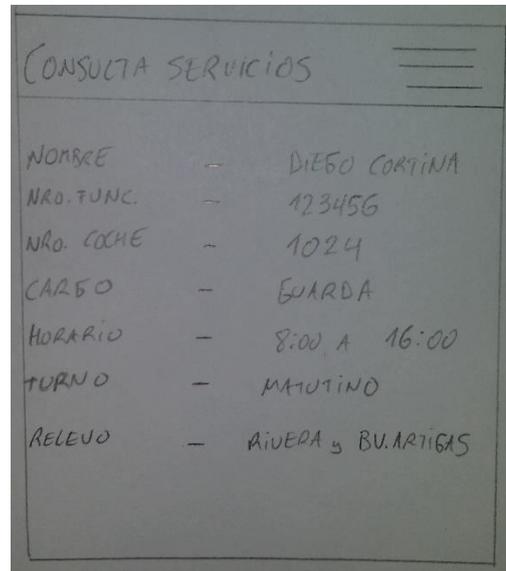


Ilustración 12-44 servicio asignado

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Servicio incorrecto”, “Servicio faltante”, “Asignación incorrecta”, “Asignación faltante”.

12.2.4.4. Casos de uso módulo Largadores

12.2.4.4.1. C4-CU1 – Modificar servicios.

RF asociado	C4-RF1
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Largador
Permisos para acceso a datos	Publico
Precondición	<ol style="list-style-type: none"> 1. Largador logueado a la aplicación. 2. El dispositivo debe tener acceso a internet. 3. Tiene que haber algún servicio dado de alta previamente.

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.

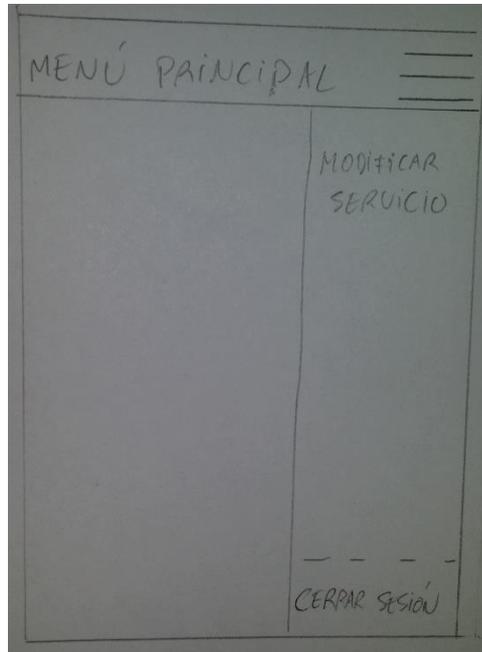


Ilustración 12-45 menú principal

2. Permitir seleccionar un servicio de una determinada fecha, el cual se quiere modificar.

MODIFICAR SERVICIO

ID SERVICIO	RECORRIDO	FECHA	
34	121	18/12/15	✓
56	142	18/12/15	

Ilustración 12-46 modificar servicio

- Se modifica la hora de salida y/o el coche asignado

MODIFICAR SERVICIO

ID SERVICIO - 34

RECORRIDO - 121

FECHA

NRO. COHE

HORA SALIDA

HORA LLEGADA

Ilustración 12-47 editar servicio

- a. Si la hora ingresada tiene formato incorrecto (ej. 12:75) se le advierte al usuario mediante un mensaje en pantalla con el texto “La hora ingresada tiene formato incorrecto”.
- b. Si la hora ingresada es invalida debido a que por reglas del negocio el coche seleccionado no pueda ser asignado en dicha hora, se le advierte al usuario mediante un mensaje en pantalla con el texto “La hora ingresada es invalida para el coche del servicio”.
- c. Si el coche ingresado es invalido (ej. 2,5) o es inexistente se le advierte al usuario mediante un mensaje en pantalla con el texto “El coche seleccionado es incorrecto”.
- d. Si el coche ingresado ya cubre otro servicio para ese horario o por alguna otra regla de negocio no puede ser asignado al servicio, se le advierte al usuario mediante un mensaje en pantalla con el texto “El coche ingresado es invalido”.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Número de coche incorrecto”, “Información de servicio incorrecto”, “Servicio faltante”.

12.2.4.4.2. C4-CU2 – Alta servicio no planificado

RF asociado	C4-RF2
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Largador
Permisos para acceso a datos	Publico
Precondición	<ol style="list-style-type: none"> 1. Largador logueado a la aplicación. 2. El dispositivo debe tener acceso a internet. 3. Deberá de haber al menos un choche (disponible para el horario especificado) y recorrido dados de alta previamente en el sistema.

Tabla 12-18 C4-CU2

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.

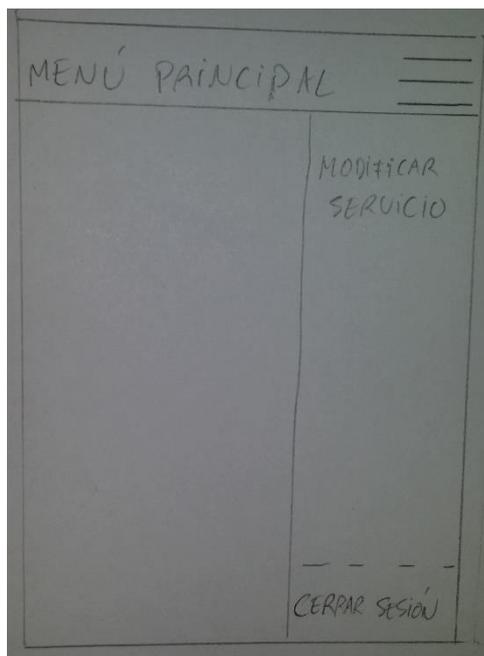


Ilustración 12-48 menú principal

2. Se ingresan todos los datos referentes al servicio a dar de alta: Nro. coche, fecha y hora de salida, fecha y hora de llegada, línea, recorrido, nro. comodín, nro. conexión.

ID SERVICIO	RECORRIDO	FECHA
34	121	18/12/15
56	142	18/12/15

Ilustración 12-49 modificar servicio

2.1 Si la hora ingresada tiene formato incorrecto (ej. 12:75) se le advierte al usuario mediante un mensaje en pantalla con el texto "La hora

ingresada tiene formato incorrecto”.

2.2 Si la hora ingresada es invalida debido a que por reglas del negocio el coche seleccionado no pueda ser asignado en dicha hora, se le advierte al usuario mediante un mensaje en pantalla con el texto “La hora ingresada es invalida para el coche del servicio”.

2.3 Si el coche ingresado es invalido (ej. 2,5) o es inexistente se le advierte al usuario mediante un mensaje en pantalla con el texto “El coche seleccionado es incorrecto”.

2.4 Si el coche ingresado ya cubre otro servicio para ese horario o por alguna otra regla de negocio no puede ser asignado al servicio, se le advierte al usuario mediante un mensaje en pantalla con el texto “El coche ingresado es invalido”.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#), se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Número de coche incorrecto”, “Información de servicio incorrecto”, “Servicio faltante”.

12.2.4.4.3. C4-C3 – Baja servicio

RF asociado	C4-RF3
RNF asociado	RNF1, RNF2, RNF3 RNF4, RNF5, RNF6, RNF7, RNF8, RNF9, RNF10, RNF11, RNF12, RNF13.
Actor	Largador
Permisos para acceso a datos	Publico
Precondición	<ol style="list-style-type: none">1. Largador logueado a la aplicación.2. El dispositivo debe tener acceso a internet.3. Debe de existir al menos un servicio dado de alta en el sistema.

Curso de la funcionalidad (con interfaz de ejemplo):

1. Seleccionar la funcionalidad desde el menú.

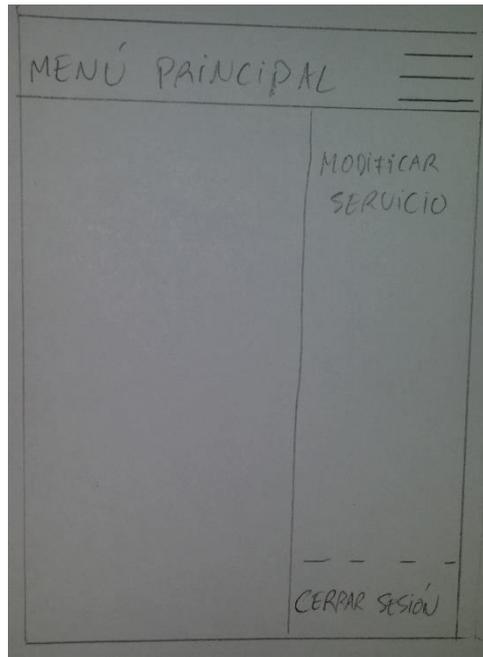


Ilustración 12-50 menú principal

- 2 Permitir seleccionar un servicio de una determinada fecha, para su eliminación.

ID SERVICIO	RECORRIDO	FECHA	
34	121	18/12/15	✓
56	142	18/12/15	

Ilustración 12-51 modificar servicio

3. Se despliega mensaje de confirmación de eliminación con el siguiente mensaje "¿Esta seguro que desea eliminar el servicio seleccionado?".

MODIFICAR SERVICIO

ID SERVICIO - 34

RECORRIDO - 121

FECHA 18/12/2015

NRO. COHE 1036

HORA SALIDA 09:15

HORA LLEGADA 11:34

GUARDAR

Ilustración 12-52 modificar servicio

4. Se elimina el servicio para la fecha seleccionada.

NOTA: Para el caso de acceder al Caso de Uso [G-CU4 Reportar Error](#) , se hará con los siguientes posibles errores como parámetros: “Error de Conexión”, “Número de coche incorrecto”, “Información de servicio incorrecto”, “Servicio faltante”.

12.3. Anexo 3 - Encuesta Inspectores

Se realizó una encuesta a un grupo reducido de 13 inspectores para poder conocer más sus habilidades con teléfonos móviles y principalmente para priorizar requerimientos. La misma se muestra a continuación:

Encuesta:

CUTCSA se encuentra planificando el desarrollo de una aplicación móvil con el objetivo de ayudarlo en sus tareas diarias como Inspector, y quisiéramos conocer su opinión al respecto. Para esto hemos diseñado una pequeña encuesta, que valoraremos mucho si se toma cinco minutos para responder.

Para las siguientes preguntas, marque con una "X" en la respuesta más adecuada.

	No tengo	iPhone	Android	Otros
¿Qué tipo de celular inteligente o tableta tiene?				
	Nunca usé	Poco hábil	Muy hábil	Soy un experto
¿Qué tan hábil se siente utilizando celulares táctiles?				
	Nunca	Pocas veces	Varias veces	Con mucha frecuencia
¿Qué tan frecuentemente alguna persona le ha hecho una consulta en la calle o en una parada? (Por ejemplo: dónde toma cierta línea; o si quiere ir a cierto lugar qué línea le sirve, etc.)				

Si tuviera una aplicación móvil instalada en su celular que le ayude en su tarea diaria, indique para las siguientes funcionalidades qué tanta utilidad le ofrecerían:

	Nada útil	Poco útil	Muy útil	Necesario
Ver las paradas más cercanas a su ubicación y las líneas que pasan por ella.				

Visualizar el recorrido de cierta línea en un mapa.				
Saber los próximos arribos de líneas a una parada.				
Ver la ubicación de cierto coche en un mapa.				
Conocer la ruta y qué ómnibus tomar desde un origen a cierto destino.				
Conocer la ubicación e información de horarios de las oficinas de venta de boletos.				
Realizar el control de boletos simplemente acercando la tarjeta del usuario al celular y que el mismo realice un sonido en caso de que la tarjeta no se haya utilizado en el viaje.				

¿Hace cuántos años se desempeña como Inspector en CUTCSA? _____

¿Se le ocurren otras funcionalidades que puedan serle útiles en su labor diaria en la empresa? Si es así, podría escribirlas en el cuadro que se encuentra a continuación.

Si le podría interesar integrar el equipo de Inspectores que realizará las primeras pruebas de la aplicación, le agradecemos que nos indique su nombre y forma de contacto:

Nombre: _____

Forma de contacto (Celular o Mail): _____

Muchas gracias!

Resultados:

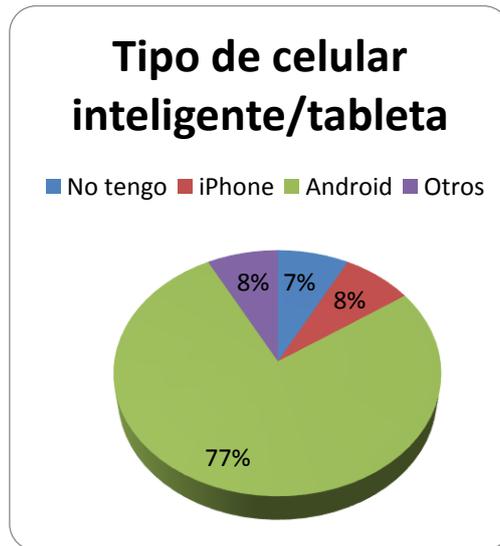


Gráfico 12-1 Tipo de celular



Gráfico 12-2 Habilidades con celulares inteligentes



Gráfico 12-3 Frecuencia de uso en la calle

Preguntas sobre requerimientos:



Gráfico 12-4 Req. información de parada

Información de recorrido

■ Poco útil ■ Muy útil ■ Necesario

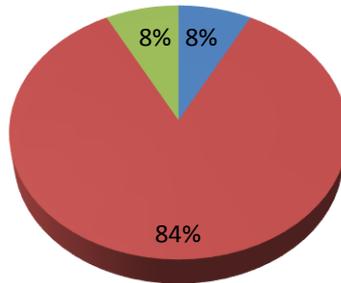


Gráfico 12-5 Req. información de recorrido

Conocer arribos de líneas a paradas

■ Muy útil ■ Necesario

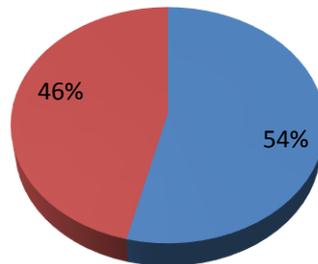


Gráfico 12-6 Req. Conocer arribos de líneas a paradas

Ubicación de coche

■ Muy útil ■ Necesario

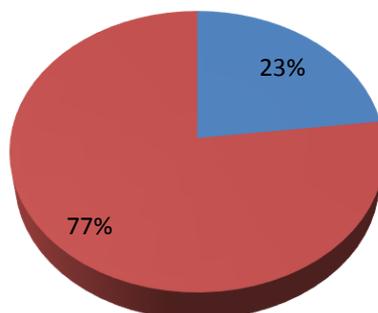


Gráfico 12-7 Req. ubicar coche

Ruta desde origen a destino

■ Poco útil ■ Muy útil ■ Necesario

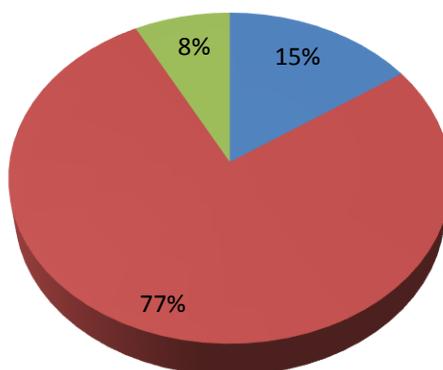


Gráfico 12-8 Req. Ruta de origen a destino

Información de oficinas de venta de boletos

■ Nada útil ■ Poco útil ■ Muy útil ■ Necesario

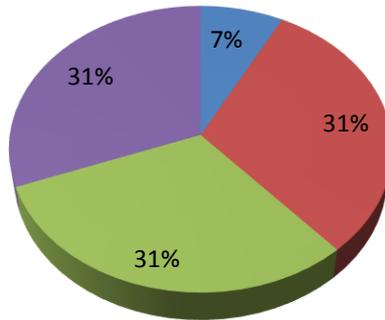


Gráfico 12-9 Req. Información de oficinas de venta de boletos

Control de servicio

■ Muy útil ■ Necesario

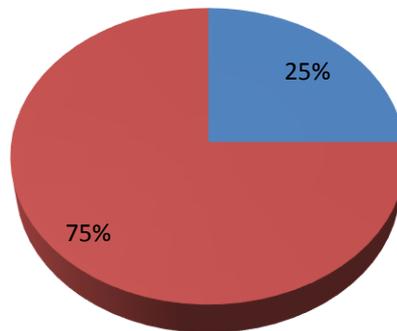


Gráfico 12-10 Req. Control de servicio

La siguiente Gráfica muestra los requerimientos priorizados por los encuestados, se asigno 0, 1, 2, 3 puntos a las respuestas: nada util, poco util, muy util y necesario, respectivamente.

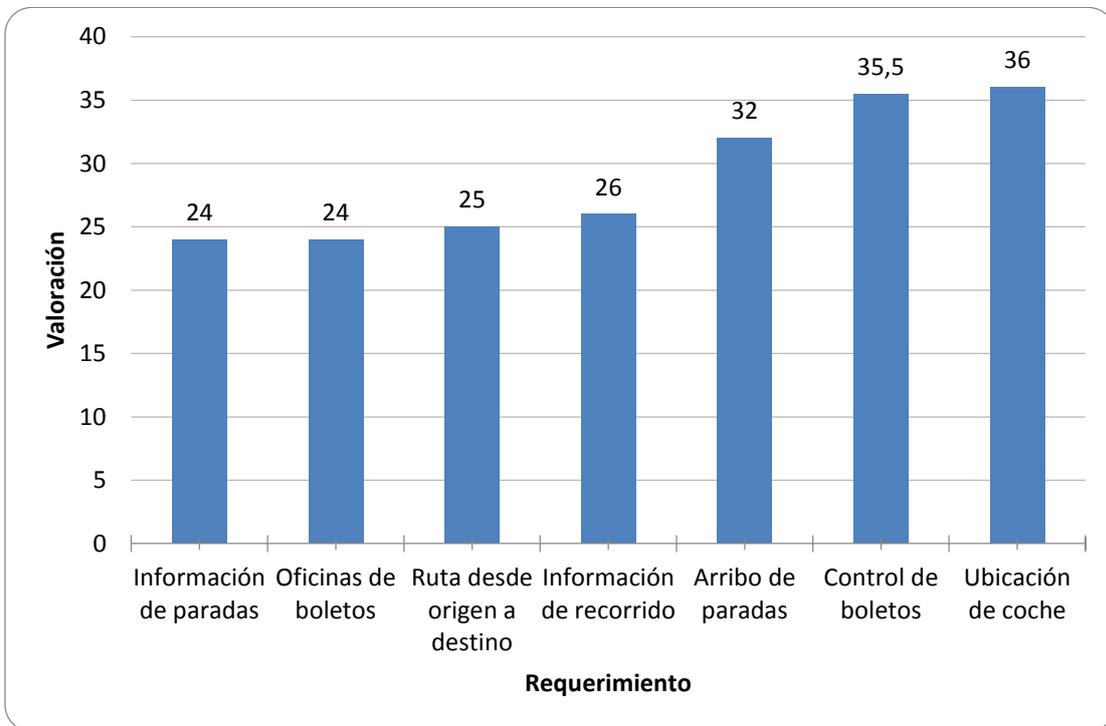


Gráfico 12-11 Priorización de requerimientos existentes.

La siguiente grafica muestra la cantidad de inspectores agrupados por cantidad de años que trabajo en la empresa.

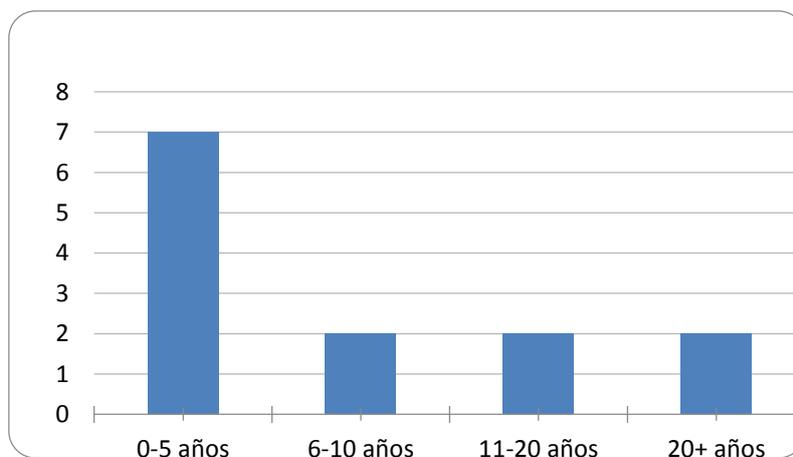


Gráfico 12-12 empleados según franja de edad de C.U.T.C.S.A.

Interés por integrar el equipo de prueba

■ No interesado ■ Interesado

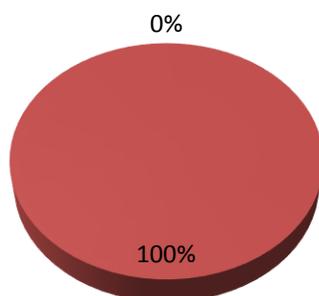


Gráfico 12-13 Interés por integrar equipo de prueba

Nuevos requerimientos propuestos		
Requerimiento propuesto	Resumen	Cantidad de veces que se solicitó
Tener conocimiento de guardia	Guardia	1
Tener conocimiento de pannes	Pannes	1
Tener conocimiento de viajes no planificados en tiempo real	Viajes no planificados	1
Tener conocimiento de desvíos planificados o alteraciones de ruta por obras, eventos, ferias, etc.	Desvíos	2
Tener conocimiento de horarios y servicios	Horarios y servicios	5
Orden de pasadas completo	Orden de pasadas	4
Contacto directo con radio o mob. 9 y 7	Contacto con radio y mob.	1
GPS para encontrar el inspector más cerca del desvío o accidente	GPS del inspector	1
Tener acceso a la lista del personal como así también saber si los funcionarios se encuentran habilitados para desempeñar las tareas eventuales	Lista del personal	1

Tabla 12-19 Requerimientos propuestas

Análisis:

De las respuestas de las entrevistas realizadas se pueden concluir varias cosas. El 77% de los encuestados utiliza un dispositivo móvil Android, lo cual es coherente con el análisis realizado en el capítulo de Arquitectura. Un 66% se considera muy hábil utilizándolo, y un 91% lo utilizan con mucha frecuencia o varias veces en el día.

En la priorización de requerimientos, se obtuvo como los más requeridos a: Ubicar coche y Control de boletos. Y por otro lado los menos requeridos: Información de paradas y Oficina de boletos.

Se obtuvieron ideas de posibles nuevos requerimientos para analizar.

Además, se categorizó a los entrevistados en franjas etarias quedando a disposición de formar parte del grupo de *testing* de la aplicación móvil en caso de ser requerido el equipo.

12.4. Anexo 4 - Documentación de Arquitectura

12.4.1. Introducción

Este documento pretende ser una breve descripción general de la arquitectura del sistema permitiéndole al lector comprender y visualizar la estructura interna del mismo. Dicha descripción será de las perspectivas de estructura, comportamiento y ambiente. Se analizará la y detallará la arquitectura del sistema desglosándola en distintos diagramas. También se detallarán las tecnologías a utilizar y las decisiones de la persistencia.

12.4.2. Propósito

El propósito fundamental de este documento será proveer al lector de una especificación completa de la arquitectura y principales decisiones de diseño involucradas en la construcción del sistema.

12.4.3. Comportamiento del Sistema

El sistema a desarrollar constará principalmente de una aplicación móvil y su respectivo *backend*, este último será el encargado de la obtención y procesamiento de los datos. En una primera instancia las funcionalidades a desarrollar van a estar enfocadas a ayudar a los inspectores, siendo la principal el control de viajes, en donde se le dará apoyo al inspector para poder validar un servicio determinado. Además de otras, localizar coches en tiempo real en el mapa, estimación de tiempo de llegada de coche real en base a información de GPS del coche, indicaciones de cómo llegar de un punto a otro en el mapa, poder ver recorrido de líneas y mostrar en mapa oficinas de venta de boletos.

12.4.4. Patrones y Principios de Diseño

En las siguientes secciones se describirán los distintos principios de diseño y patrones, que en el caso de haber sido utilizados se adjuntará además una justificación del porqué y cómo se hizo. Y para el caso de que se haya decidido no utilizar, se detallará el porqué de esta decisión y/o por qué no se estarían respetando.

12.4.5. Principios

12.4.5.1. A nivel de paquetes

Equivalencia Reúso Liberación (REP): Si bien se va a hacer un seguimiento de versiones y utilizar un sistema de control de cambios, dada la extensión del sistema y que se estima que los distintos módulos del sistema no van a tener un gran reúso de código entre sus funcionalidades dada la arquitectura elegida para la aplicación para el desarrollo del *backend*. Para el *frontend* se utilizan los paquetes Gestión UI y Elementos UI.

Dependencias acíclicas (ADP): Como se puede ver en la sección de Diagramas de paquete, tanto el diagrama de paquetes de la aplicación móvil como la del *backend* la estructura de dependencia entre paquetes forma un grafo dirigido y acíclico, por lo que se está respetando el principio de diseño.

Por otro lado, dentro de la aplicación móvil, sí existe una dependencia cíclica usada para disminuir la complejidad de la solución. Dicha dependencia se encuentra asociada al GestorMenu, la cual es llamada desde los demás paquetes de la aplicación (incluyendo la InterfazComún y los distintos Módulos), y se encuentra relacionada a las actividades (*Activity*) de la aplicación, las cuales se encuentran dentro de los paquetes de InterfazComún y los Módulos. Por dicha razón hay que tener una atención importante al modificar las clases asociadas.

Principio de Clausura Común (CCP): Las clases candidatas a cambiar por un mismo motivo fueron agrupadas en un mismo paquete para poder así minimizar el impacto del cambio, en la sección de Diagramas de paquete y Diagramas de clase se puede ver con mayor detalle cómo se agruparon las clases y corroborar la correcta aplicación del principio.

Principio de Reúso Común (CRP): Dado que se trató de respetar el principio de clausura común y las clases candidatas a cambiar por un mismo motivo están juntas, cuando reutilizamos un paquete vamos a estar utilizando la totalidad de las clases del mismo, ya que estas clases están muy relacionadas entre sí. Por lo que podemos concluir que se respeta y aplica este principio de diseño.

Principio de dependencias estables (SDP): Como se puede ver en la sección de Diagramas de paquete, tanto el diagrama de paquetes de la aplicación móvil como la del *backend* los paquetes solo dependen de paquetes más estables que ellos mismos.

En el diagrama de la aplicación móvil, el paquete “Interfaz” por ser el que contiene todo lo relacionado a la interfaz gráfica va a ser un paquete bastante inestable debido a las distintas modificaciones que la misma pueda sufrir. Dicho paquete depende del paquete

“Dominio” el cual contiene entidades del sistema necesarias para la aplicación móvil y de “AccesoAServicios”, paquete que agrupa la gestión de la conexión y acceso a datos de los servicios que van a alimentar la aplicación móvil. Finalmente hay una dependencia del paquete “AccesoAServicios” hacia “Dominio”, siendo este último el paquete más estable de todos, ya que se espera que los servicios no cambien, sino más bien que los mismos sean extendidos a futuro.

En el diagrama del *backend*, el paquete “ServiciosWeb”, el cual agrupa los servicios web que se brindarán a la aplicación móvil, depende del paquete “LogicaServidor”, paquete que contiene toda la lógica relacionada al *backend*. A su vez este último depende del paquete “AccesoADatos”, el cual se encarga de la conexión y gestión de datos con la Base de datos. Por último, los 3 paquetes mencionados anteriormente dependen de un paquete llamado “VO” el cual contiene clases utilitarias que son utilizadas por varios módulos del sistema, las cuales son muy poco probable que cambien, sino más bien que sean extendidas, haciendo de este paquete el más estable de los 4.

Debido a todo lo comentado anteriormente podemos afirmar que se respeta y aplica este principio de diseño.

Abstracciones Estables (SAP): El paquete “AccesoAServicios” en el diagrama de paquetes de la aplicación, si bien sufrirá cambios debido a la naturaleza extensible de la aplicación, estos cambios no implicarán modificaciones en las funcionalidades ya existentes y contarán con interfaces definidas para comunicación con el resto de la aplicación por lo que se le considerará un paquete estable.

En cambio, en el mismo diagrama el paquete interfaz contará con clases concretas y muy poco abstractas que implementarán la interfaz de la aplicación, por lo que se le considerará un paquete inestable.

En el [diagrama de paquetes del backend](#), el paquete de “AcceseADatos” es un paquete estable que solo se modificará para extender la aplicación. Lo mismo pasará con el paquete “ServiciosWeb”. El paquete “LogicaServidor” es concreto debido a que en él están las implementaciones de las funcionalidades del *backend*. Esto hace que este principio no se cumpla debido a la relación de dependencia entre los paquetes de “ServiciosWeb” y “LogicaServidor”, pero para mantener el principio desde los paquetes de la aplicación se decidió por violar el mismo en este diagrama.

12.4.5.2. Principios a nivel de clase

Principio de responsabilidad única (SRP): Las clases identificadas en los diagramas de clases de la aplicación móvil tienen claramente identificadas sus funciones. Las clases dentro de los paquetes de los diferentes módulos, son las actividades que se corresponderán con una pantalla, y esa pantalla corresponderá con una funcionalidad,

es decir tiene una sola razón por la cual cambiar. Lo mismo pasa con las clases de los paquetes “InterfazComun” y “GestoresUI”. El paquete “ElementosUI” tiene clases genéricas para mostrar información por lo que tienen definida su responsabilidad. Las clases del paquete de “AccesoADatos” tienen como fin obtener los datos desde los servicios y sólo cambiarán las clases para obtener otro tipo de información.

Las clases del *backend* están pensadas para que funcionen para cada módulo de forma independiente. Cada clase tiene la responsabilidad de responder a un único módulo del sistema.

La clase identificada para el intermediario no cumple con este principio debido a que un nuevo módulo o una modificación en cualquier requerimiento puede implicar su modificación.

Las entidades software deben estar abiertas para su extensión, pero cerradas para su modificación. La clave son las abstracciones: Polimorfismo físico (Dinámico), Polimorfismo estático (*Templates, Generics*). Hay que depender de abstracciones y no de elementos concretos.

Inversión de dependencias (DIP): En los diagramas de clases se puede apreciar que las clases de pertenecen a los módulos de alto nivel no dependen de aquellas clases que implementan funcionalidad de bajo nivel

Segregación de interfaces (ISP): Se pueden identificar solamente una interfaz en el diseño correspondiente al paquete de clases de la aplicación móvil. Esta solo tiene una clase la cual la implementa.

12.4.6. Patrones

Observer: Dentro de la aplicación móvil, se utilizó dicho patrón para poder actualizar el estado de una actividad cuando la misma llama a un servicio. Cada servicio (usado para realizar una consulta al servidor) debe ejecutarse en un nuevo *Thread*, y para notificar a la actividad (*Activity*) el estado del mismo (que entre los mismos se encuentran la realización de la llamada al servidor, el fin correcto de la llamada, o el error en la llamada), se debe pasar por parámetro un *IAccesoAServiciosCallback*, el cual es creado por la actividad (*Activity*) que llama al servicio, definiendo qué hacer cuando el servicio actualizó su estado (en *onActualizar*), cuando finalizó correctamente el servicio (en *onFinalizar*), o existió un error durante la llamada al método(en *onError*).

Para tener una interfaz genérica (y de esa manera no tener que crear muchas clases similares), el método *onFinalizar* recibe como parámetro u *Object*, que es algo que se quiere pasar desde el Servicio a la actividad (*Activity*) que lo llamó. Luego, como la

actividad (*Activity*) sabe qué es lo que está recibiendo, realiza un *casting* del objeto obtenido por parámetro en el *onFinalizar*.

Se decidió usar dicha solución por distintas razones:

- Como la llamada al servidor debe realizarse en un nuevo *Thread*, no puede retornar un valor en el *return* del método del Servicio.
- Como la llamada al servidor debe realizarse en un nuevo *Thread*, no puede pasarse por parámetro en el método del Servicio la referencia al objeto que se quiere obtener como resultado de la llamada al Servicio, dado que el mismo para ser usado dentro del *Thread* debe ser final.
- Cada actividad (*Activity*) puede recibir distintos tipos de datos desde la llamada al método del servicio.
- Reduce la cantidad de interfaces a crear, dado que no se necesita crear una interfaz por cada tipo de objeto que se desea obtener desde el servicio a través del *onFinalizar*.

Singleton: Las clases Sistema, GestorLocalizacion, y GestorMenu implementan este patrón dado que son clases que encapsulan valores necesarios en memoria, y sólo se necesitan 1 única instancia de las mismas.

Fachada: La clase GestorMenu se usa como puntos de acceso a las distintas opciones del menú lateral, y así abrir las distintas actividades (*Activity*) asociadas a cada opción.

Command: Tal como se explicaba anteriormente con el *IAccesoAServiciosCallback*, cuando se llama al Servicio desde una actividad (*Activity*), dicha actividad (*Activity*) crea un new *IAccesoAServiciosCallback* que encapsula qué acción realizará en el *onFinalizar*, *onActualizar*, y *onError*, se pasa dicho objeto por parámetro al Servicio (que realiza la llamada al servidor) y ejecuta los distintos métodos del *IAccesoAServiciosCallback* concreto dependiendo de los resultados de la llamada al servidor.

Abstracciones: Las clases *MapaActivity*, *MenuActivity*, y *ComunActivity* definen comportamientos comunes a todas las clases que hereden de ellos, pero a su vez las clases que los heredan puede redefinir o extender los métodos para realizar sus propias soluciones en cada caso que se necesite a través del *@Override*, y *super.metodo()*.

- Cada actividad (*Activity*) de la aplicación debe heredar de *ComunActivity*, ya que define en el *onCreate()* y *onDestroy()* el manejo del *stack* de actividades (*Activity*) abiertas durante el ciclo de vida de la aplicación.
- Toda actividad que quiera mostrar el menú lateral, debe extender de *MenuActivity*, ya que define toda la funcionalidad común referente al menú lateral. Si no se quiere mostrar el menú lateral, extender de la clase *ComunActivity*.

- Toda actividad que desee mostrar un mapa por pantalla (y permitir el acceso al menú lateral), debe extender de `MapaActivity`, ya que define todo el comportamiento de los marcadores, zooms, movimientos de cámara, y otros referentes al mapa. Cada vez que se desea cambiar la funcionalidad de cierto método referente al mapa, se puede realizar un `@Override`. Cada vez que se desea extender la funcionalidad de cierto método, llamar al `super.metodo()` dentro del método y (antes o después) agregar la funcionalidad.

Memento: No es exactamente la aplicación de dicho patrón, pero sí se utiliza el almacenado del *stack* de actividades (*Activity*) que se fueron abriendo en la aplicación para tener una gestión de los mismos. Se usa en el caso de abrir una actividad de cierta funcionalidad que ya se había abierto previamente. Dicho manejo del *stack* se realiza en la clase `Sistema`.

12.4.7. Tecnología a utilizar

Se utiliza para el desarrollo del *backend* C# en .net utilizando Visual Studio y para el desarrollo de la aplicación móvil java utilizando el Android Studio.

Como repositorio se utiliza Git, y Bitbucket como cliente.

Para versionar y guardar tanto los documentos del proyecto como cualquier otro archivo necesario para el proyecto se utiliza Google Drive en una carpeta compartida.

12.4.8. Diagramas

12.4.8.1. Capas Lógicas

En esta arquitectura se propone un diseño basado en el estilo de capas lógicas las cuales proveen diferentes niveles de abstracción.

Con esto se logra que cada componente cuente con un conjunto finito de funcionalidades y responsabilidades.

La selección de la arquitectura se basó en las buenas prácticas de diseño favoreciendo la mantenibilidad y modificabilidad del sistema.

La separación en capas lógicas permite minimizar el acoplamiento entre los componentes dado que cada componente se relaciona con otro con el mismo o menor nivel de abstracción.

Los diferentes componentes de las capas exponen sus funcionalidades a través de interfaces bien definidas. Con esto se logra exponer únicamente las funcionalidades necesarias y se ocultan los mecanismos internos de cada componente favoreciendo el reúso de estos y la modificabilidad del sistema. A su vez, esta agrupación en los diferentes niveles de abstracción permite que a futuro dichas capas puedan ser distribuidas físicamente modificando únicamente los mecanismos de comunicación.

El lado negativo del diseño propuesto es que los cambios de comportamiento generados en los diferentes componentes pueden generar un impacto en las otras capas dada la propagación de dicho cambio.

A continuación, se detalla la arquitectura propuesta y sus componentes:

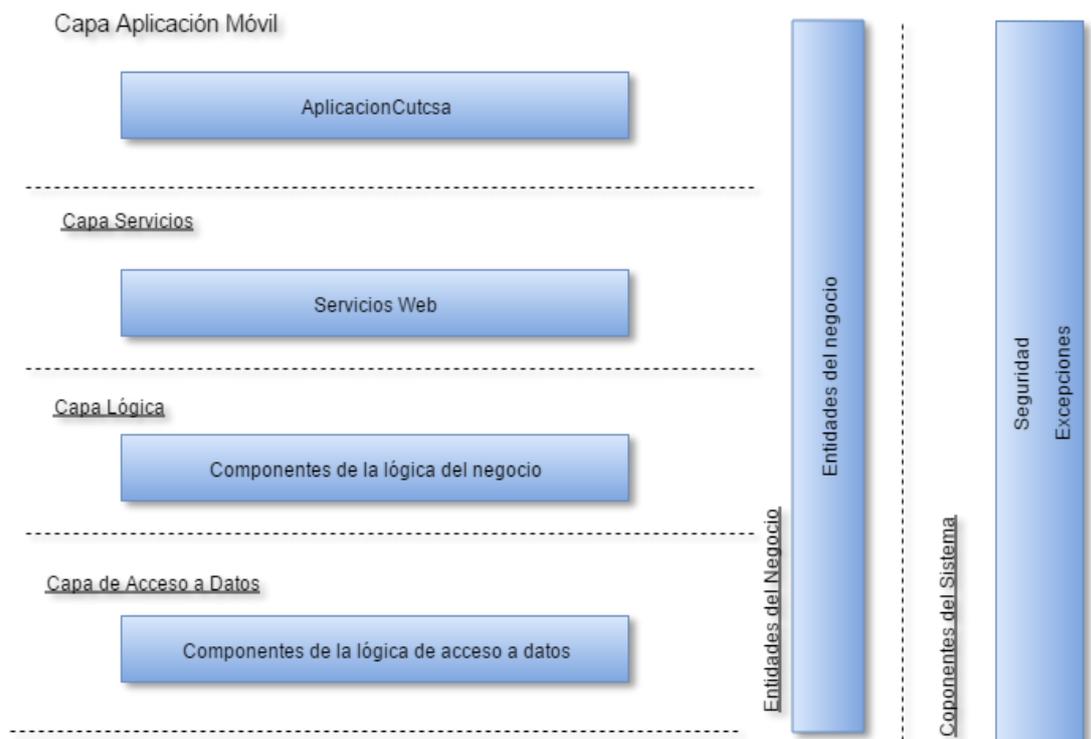


Ilustración 12-53 Capas Lógicas

Como se comentó anteriormente, se pueden apreciar claramente las diferentes capas de la aplicación las cuales son:

- Capa Aplicación Móvil: Dentro de esta capa se encuentran los componentes dedicados a la interfaz gráfica de la aplicación, así como el manejo de la comunicación con los servicios.
- Capa Servicios: Esta capa es la encargada de publicar los servicios que se utilizarán en la aplicación. web api .net

- Capa Lógica: Dentro de esta capa se encuentra la lógica de la aplicación, así como también las diferentes reglas del negocio.
- Capa de Acceso a Datos: Esta capa es la encargada de proveer de datos a la aplicación.
- Capa de Intercambio de Datos: La responsabilidad de los componentes de esta capa es la comunicación entre la base de datos de C.U.T.C.S.A. y la base de datos que se utilizará en la aplicación.
- Capas transversales:
 - Entidades de negocio: La capa de entidades de negocio provee a la aplicación de los objetos necesarios para la representación correcta de las entidades del dominio.
 - Componentes del sistema: Dentro del sistema existen diferentes requerimientos y funcionalidades las cuales se han agrupado en componentes que resuelven dichos casos. Entre ellas se encuentra las excepciones, y seguridad.

12.4.8.2. Diagrama de Deployment

La arquitectura de despliegue fue sugerida por el cliente, la cual fue pensada en la seguridad del sistema. Una de las características fundamentales de la misma es el hecho que la solución que se desarrolló esta desacoplada de los sistemas existentes. Si bien la solución depende de estos para el acceso a datos, el desarrollo se realizó de forma independiente teniendo en cuenta los atributos de calidad mencionados anteriormente. Es decir que, si bien existieron restricciones, también hubo libertad de utilizar cualquier otra tecnología que fuese necesaria.

La otra característica fundamental de esta arquitectura es que fue pensada en la seguridad de los datos. Los datos en la base de datos de C.U.T.C.S.A. son muy sensibles y actualmente no se pueden acceder sin ser dentro de la red local. Mediante una nueva base de datos del proyecto que solo tendrá información relacionada con la aplicación, se podrá proteger la base de datos principal de la empresa. Si bien algunos de los datos pueden contener información sensible, solo se estaría trabajando con algunos de ellos, no todos los que existen en la base de datos actual.

Para obtener los datos de la empresa, se sugirió la creación de un intermediador que cumplirá el rol de volcar los datos necesarios de una base a otra. Este solo podrá leer datos de la base de datos de C.U.T.C.S.A y guardar en la base de datos del proyecto. Esta operación se realiza por la red local. De esta forma en caso de existir un ataque al sistema y en el caso que puedan acceder a la base de datos del proyecto, no tendrán acceso a ningún otro nivel de datos más que en el que se encuentran. De todas formas,

como se mencionó anteriormente, los datos más sensibles se encuentran cifrados y las operaciones con los mismos se realizan por medio de SSL.

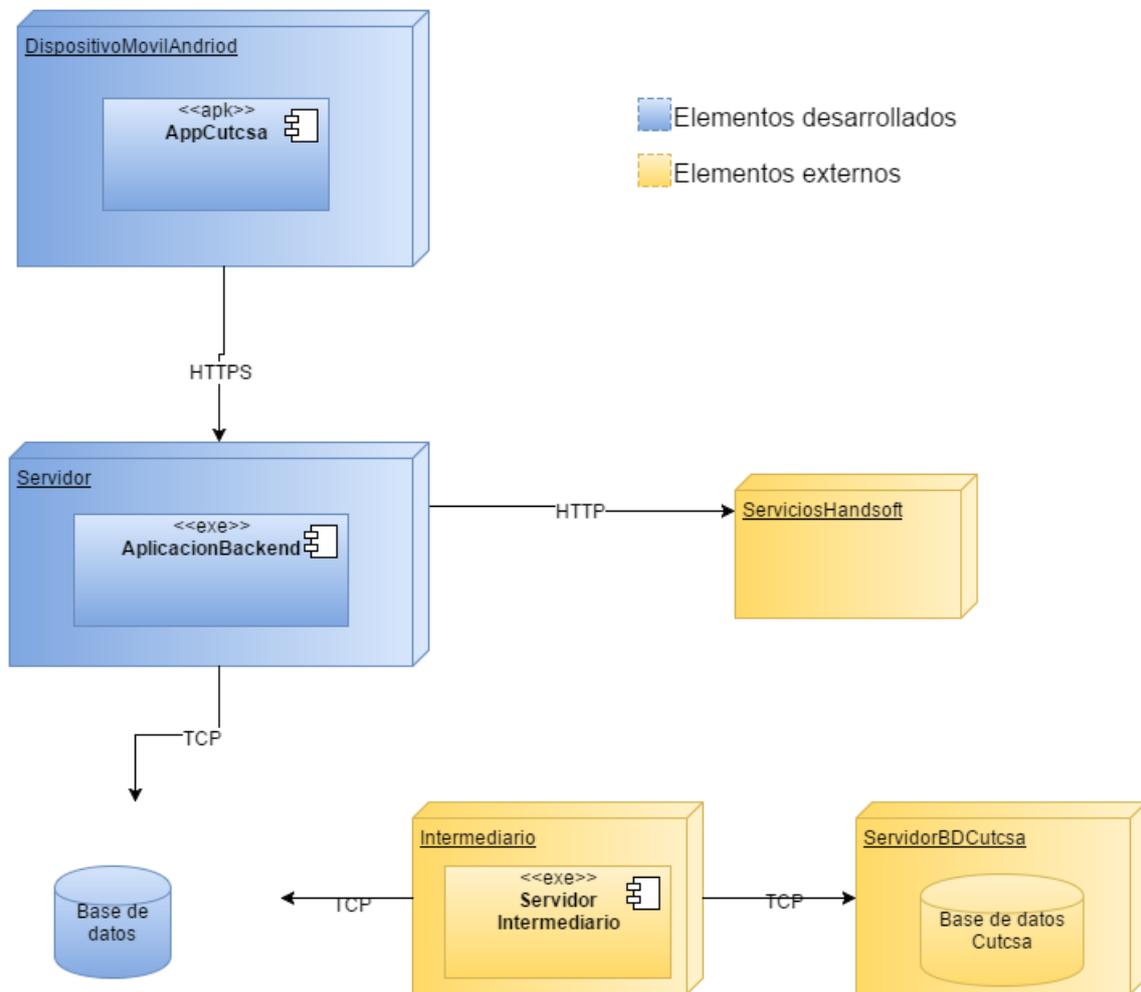


Ilustración 12-54 Diagrama de deployment

Elemento	Responsabilidad
DispositivoMovilAndriod	El Dispositivo Móvil Android es en donde se encuentra la aplicación que utilizará el usuario. Este tendrá las funcionalidades relevadas.
ServidorBD	Contiene la base de datos del proyecto con la cual la aplicación obtendrá los

	datos. Solo tendrá los datos que van a usarse en la aplicación.
ServiciosHandsoft	Los servicios de Handsoft contienen la información que se utiliza en la aplicación relacionada a la posición de los coches y el tiempo de demora de un coche para trasladarse de un punto a otro.
Intermediario	Intermediario entre la base de datos de C.U.T.C.S.A. y la base de datos que accederá la aplicación <i>backend</i> , cuyo objetivo es mantener la seguridad de los datos de la primera y actualizar la base de datos del proyecto.
ServidorBDCutcsa	Este componente no se modificará, solo se obtendrán los datos necesarios del mismo para volcarlos luego en la base de datos del proyecto.

Tabla 12-20 elementos de deployment

12.4.8.3. Diagrama de Componentes

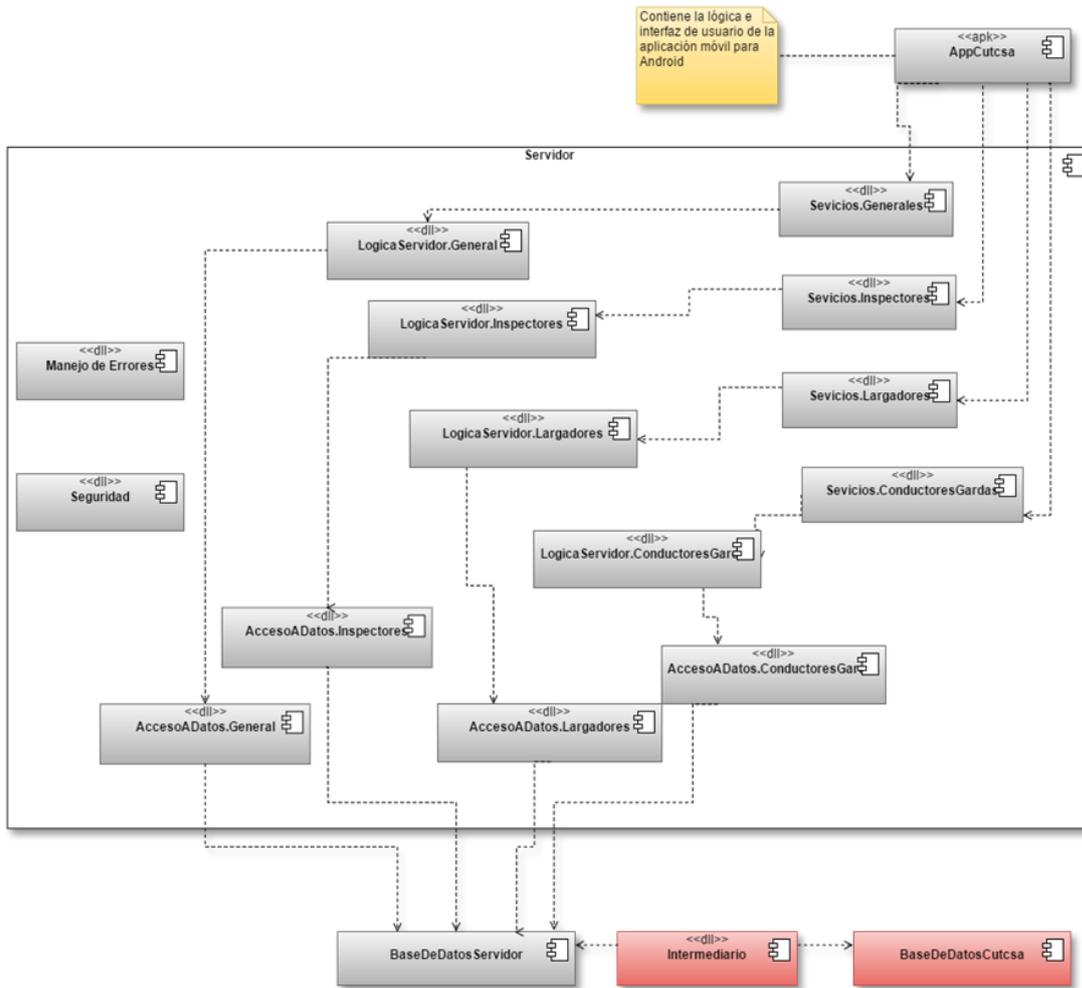


Ilustración 12-55 Diagrama de componentes

En la ilustración 6.4 se puede apreciar los distintos componentes del sistema. En rojo se encuentran aquellos que no fueron desarrollados por el equipo. Los componentes de Seguridad y Manejo de Errores están representados como si no tuviesen ninguna relación con el resto de los componentes, pero es así para simplificar el diagrama ya que de estos dependen todos los componentes del *backend*.

La estructura del componente servidor se diagramó de esta forma ya que en C.U.T.C.S.A. las aplicaciones .net están hechas con esta estructura. La decisión fue tomada para conservar la misma solución, ya que la aplicación va a ser mantenida eventualmente por la empresa.

Elemento	Responsabilidad
AppCutcsa	Aplicación que utilizará el usuario

Servicios.Generales	Brindar los servicios que se utilizarán en la aplicación para funcionalidades comunes a todos los módulos
Servicios.Largadores, Servicio.Inspectores, Servicios.ConductoresGuardas	Brindar los servicios que se utilizarán en la aplicación. Cada uno de estos componentes sirven únicamente para el módulo para el cual fueron creados.
LogicaServidor.General	Manejar la lógica y los datos que se utilizarán en la aplicación para las funcionalidades comunes a todos los módulos
LogicaServidor.Largadores, LogicaServidor.Inspectores, LogicaServidor.ConductoresGuardas	Manejar la lógica y los datos utilizarán en la aplicación. Cada uno de estos componentes sirven únicamente para el módulo para el cual fueron creados.
ManejoDeErrores	Realizar validaciones comunes y del manejo de excepciones del <i>backend</i>
Seguridad	Tiene la lógica correspondiente a la solución de seguridad y manejo de sesión.
AccesoADatos.Largadores, AccesoADatos.Inspectores, AccesoADatos.ConductoresGuardas	Traer y guardar datos de la base de datos para las funcionalidades específicas de cada módulo para el cual fueron creados.

Tabla 12-21 Elementos de componentes

12.4.8.4. Diagramas de Paquetes

12.4.8.4.1. Paquetes de la aplicación Móvil

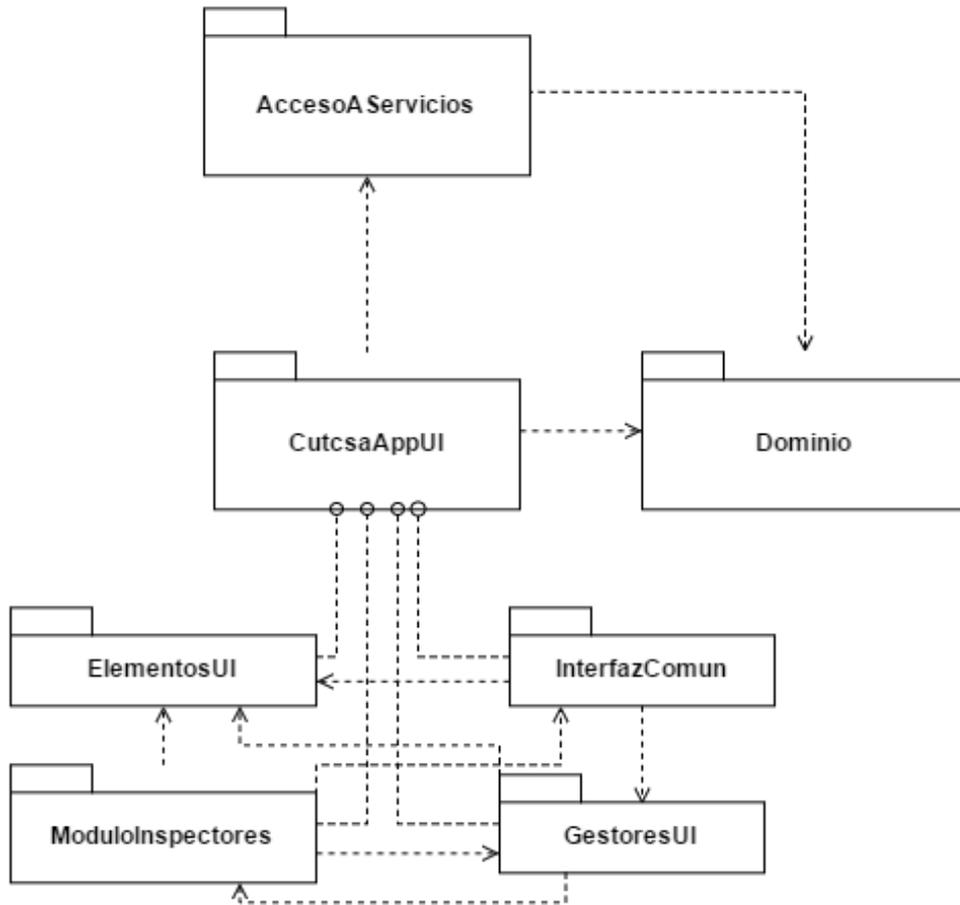


Ilustración 12-56 Diagrama de Paquetes Aplicación

El paquete “AccesoAservicios” es el responsable de, como lo indica su nombre, acceder a los servicios que se utilizarán en la aplicación. En este paquete los servicios son empaquetados en objetos. Éste se comunica con “Dominio” para identificar la información obtenida en los objetos del sistema.

El paquete “Dominio” cuenta con los objetos y clases con los que se trabajarán en la aplicación.

“CutcsaAppUI” es el paquete responsable de la interfaz de la aplicación. Se comunica con “AccesoAservicios” y “Dominio” para obtener la información que desea mostrar.

En el paquete “ModuloInspectores” se encuentran las actividades que se utilizarán para mostrar los requerimientos relacionados a los inspectores.

El paquete “ElementosUI” tiene como función ofrecer de base para mostrar diferentes estructuras de datos en pantalla. Esto es por ejemplo mostrar una lista o imágenes personalizadas. Esto permite encapsular mejor las funcionalidades y el reuso de código.

“GestoresUI” Tiene como función de agrupar todas las funcionalidades que hacen referencia a la gestión dentro de la aplicación (específicamente de elementos de la interfaz de usuario), que incluyen mostrar elementos por pantalla al usuario, gestión del acceso a opciones del menú, etc.

El paquete “InterfazComun” se utiliza para mostrar la funcionalidad común a todos los módulos que tendrá el sistema. Un ejemplo es el log in de la aplicación. También cuenta con actividades reutilizables por los módulos como ser el Mapa y el menú

12.4.8.4.2. Paquetes del Backend

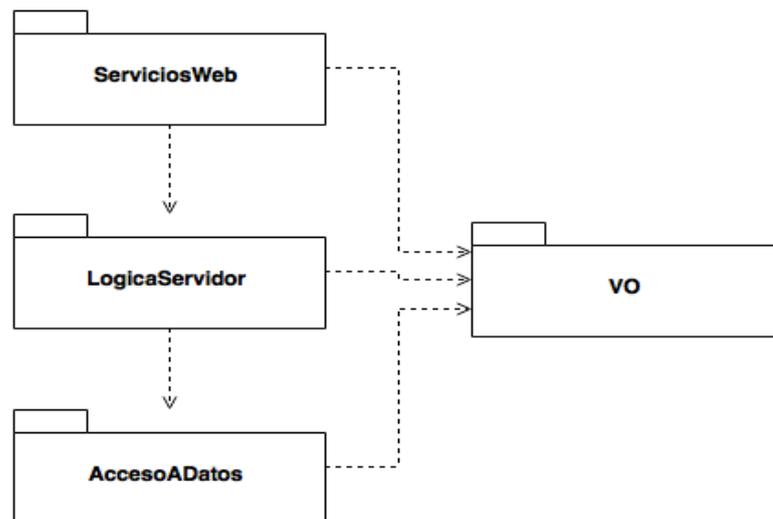


Ilustración 12-57 Diagrama de paquetes Backend

Los componentes descritos en la sección anterior, se encuentran dentro de los paquetes de este diagrama. Los componentes del tipo “Servicios.Componente” están dentro del paquete “ServiciosWeb”, los del tipo “LogicaServidor.Componente” se encuentran en el paquete “LogicaServidor” y de la misma forma los componentes “AccesoADatos.Componente”, están dentro del paquete “AccesoADatos”.

Dada la granularidad de los componentes, estos no tienen paquetes dentro de ellos, solo clases.

El paquete “SeviciosWeb” es el que contiene la lógica en los que respecta a la publicación de los servicios que serán utilizados en la aplicación. Estos servicios no solo serán utilizados para obtener información desde el *backend*, sino también para enviar información desde la aplicación. Algunos de los servicios a utilizar se obtendrán desde los servicios publicados por la empresa Handsoft, dado que hay información y funcionalidades que ya están desarrolladas por la empresa y que actualmente no es posible desarrollar. Esto se realizó de esta forma ya que, si luego se logra realizar el desarrollo de la funcionalidad desde el *backend*, la aplicación no debe de actualizarse ya que los servicios expuestos serán los mismos.

El paquete “LogicaServidor” contiene la lógica de la aplicación *backend* para cada módulo.

El paquete “AccesoADatos” contiene la lógica de acceso a datos para la aplicación *backend* para cada módulo.

El paquete “VO” tiene como objetivo contener los objetos comunes al sistema.

12.4.8.5. Diagramas de Clases

12.4.8.5.1. Diagrama de clases de la aplicación móvil

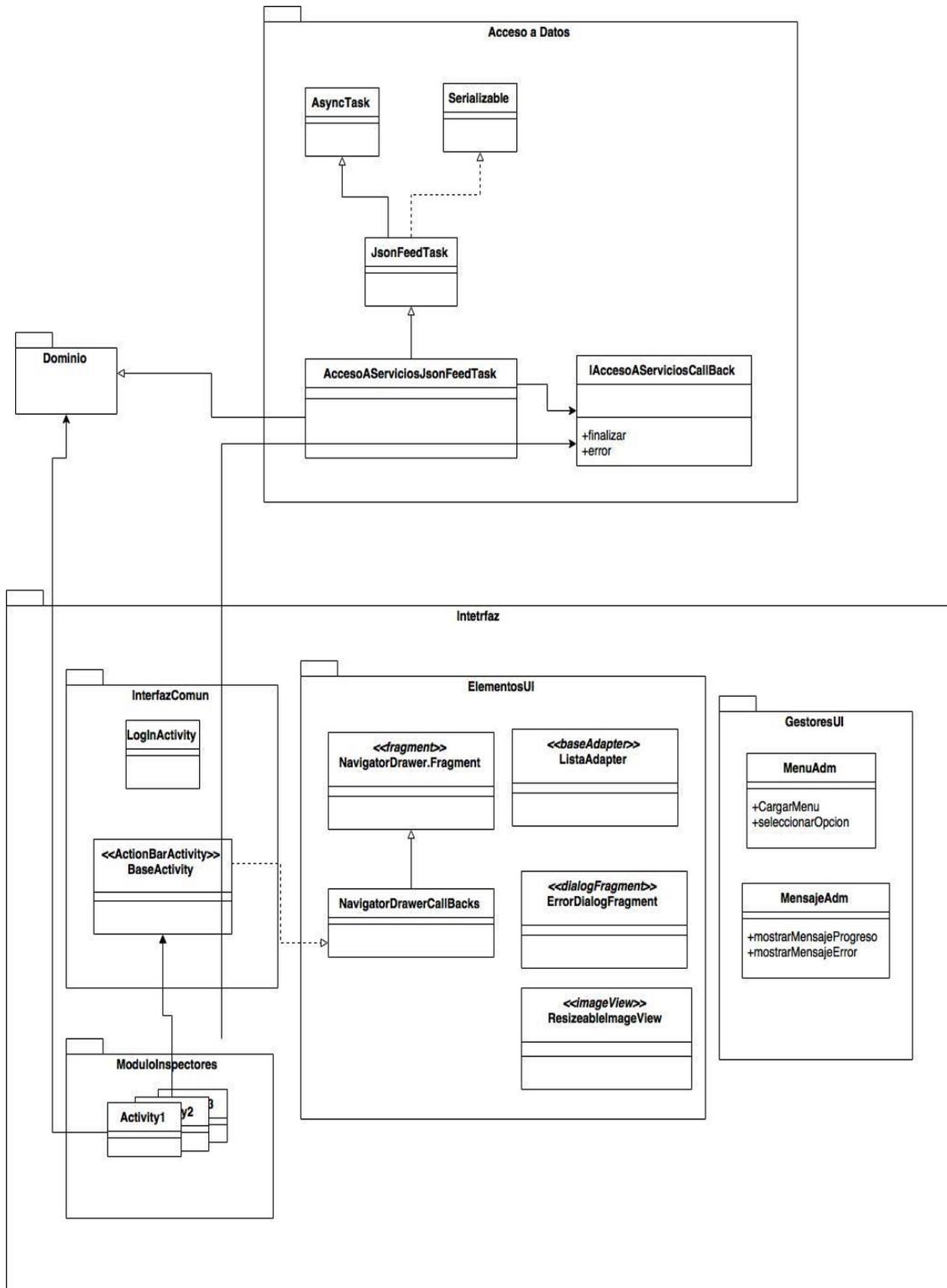


Ilustración 12-58 Diagrama de clases aplicación

En el diagrama se pueden apreciar cuáles serán algunas de las clases con las que contará la aplicación móvil y cómo se relacionan entre ellas.

Las clases del paquete “ElementosUI” son utilizadas para mostrar información en pantalla y están pensadas para poder ser reusadas desde cualquier actividad. Por ejemplo, la clase “ErrorDialogFragment”, es utilizada para los distintos mensajes de error.

Las clases del paquete “InterfazComun”, son las actividades comunes a todos los módulos, como lo es por ejemplo la clase del log in de la aplicación “LoginActivity”. También son actividades genéricas reutilizables en las distintas actividades de los módulos, como ser el menú lateral y el mapa.

Las clases de los paquetes de los módulos son las actividades correspondientes a cada módulo. Al separar las mismas en paquetes se logra aislamiento y cohesión entre las mismas.

Las clases del paquete “AccesoAServicios” cuentan con 2 propósitos, uno es acceder a los servicios correspondientes, y otro para transformar la información obtenida en objetos del sistema.

12.4.8.5.2. Diagrama de clases del servidor

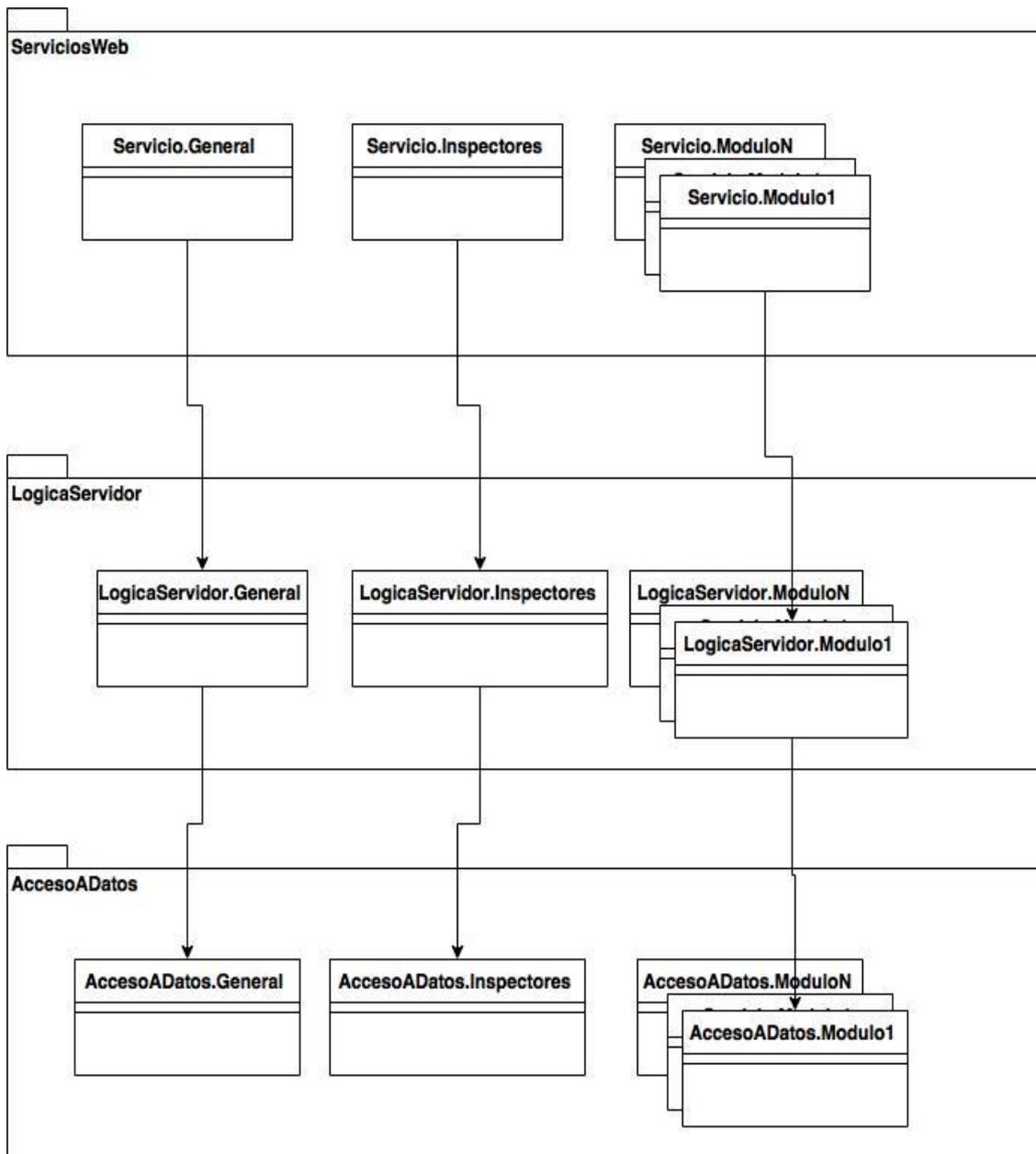


Ilustración 12-59 Diagrama de clase backend

Se puede apreciar en el diagrama que las clases coinciden con nombres de los componentes. Las responsabilidades de las clases son las mismas que las de su componente correspondiente. Las clases que no pertenecen al mismo módulo no se comunican entre ellas, de esta forma la aplicación es extensible y si se agregan nuevos módulos, el resto de los módulos no se ven afectados.

12.4.8.5.3. Diagrama de clases del intermediario

El intermediario cuenta solamente con una clase principal. Esta es la encargada de pasar los datos de la base de datos de C.U.T.C.S.A. a la base de datos que se utiliza en la aplicación.

Esta clase, aprovechando las herramientas que ofrece C#, se divide en regiones. Esta es la habilidad que brinda C# de agrupar código de una clase en las llamadas regiones. Estas comienzan con el encabezado *#region NombreRegion* y termina con *#endregion*. Cada región corresponde a una tabla en la base de datos, de forma que, si se quiere obtener un nuevo atributo de la tabla, solamente se debe modificar esa región en la clase.

Si se desea agregar una nueva tabla, se creará una región específica para la misma en la clase.

12.4.8.5.4. Diagramas de Secuencia

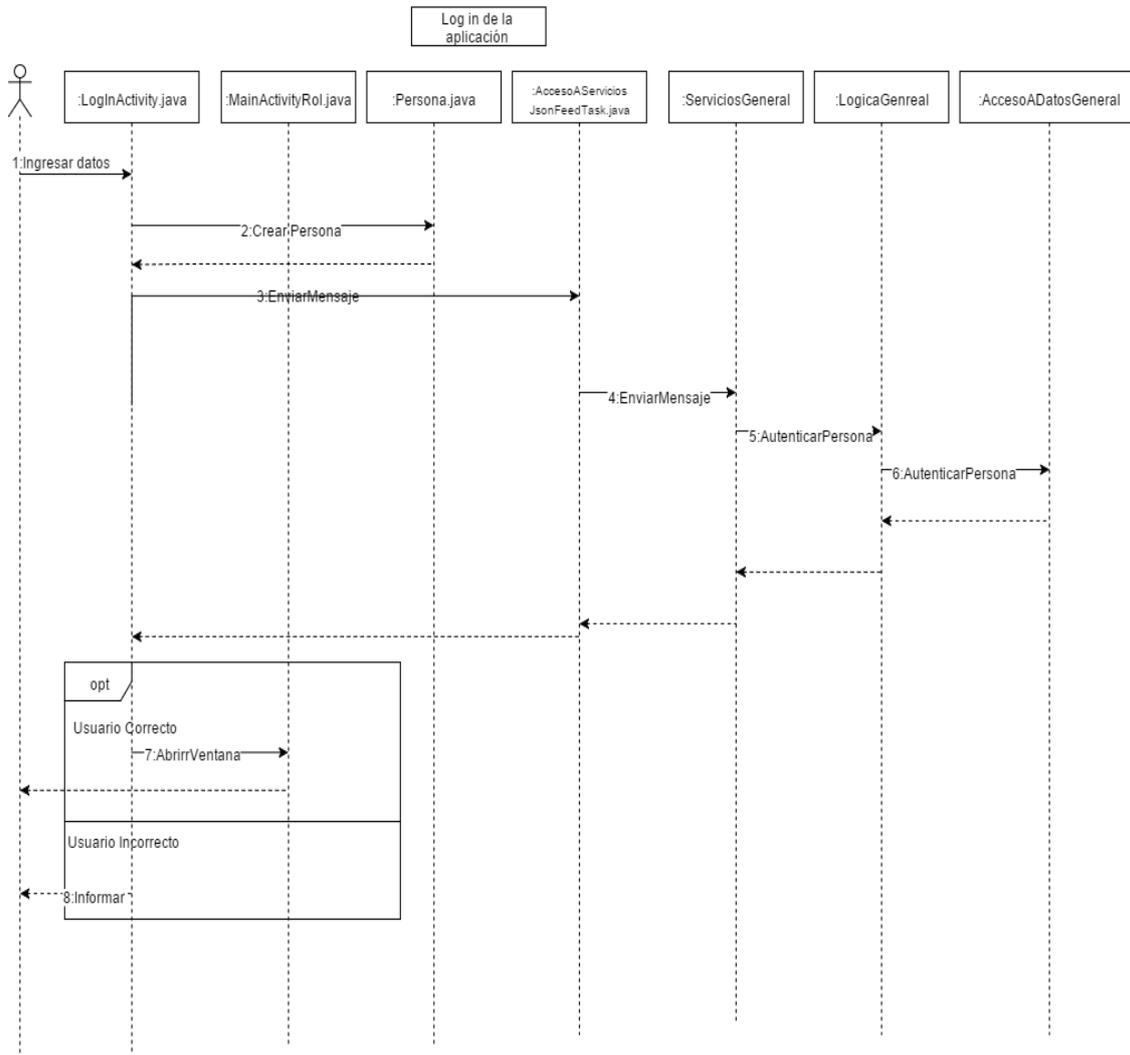


Ilustración 12-60 Diagrama de secuencia

El diagrama anterior muestra un ejemplo de la acción de *log in* en la aplicación.

12.4.9. Descripción de la Persistencia del sistema

Para la persistencia del sistema se optó por crear una base de datos auxiliar a la base de datos que utiliza C.U.T.C.S.A. Esto se realizó para mantener la integridad de los datos y mantener la seguridad de los mismos. Gracias a esta decisión, la base de datos de C.U.T.C.S.A. no sufrirá ninguna modificación por parte de la solución propuesta, simplemente se utilizará la información relevante para la solución de la misma.

Para comunicar ambas bases, como se ha explicado anteriormente en el documento, se utiliza un intermediario. Este actuará siempre dentro de la red de C.U.T.C.S.A. haciendo imposible el acceso al mismo a través de la aplicación.

12.4.10. Cumplimientos con los Atributos de Calidad

A continuación, se describen los principales requerimientos no funcionales y los atributos de calidad más relevantes tomados en cuenta en la arquitectura.

12.4.10.1. Fiabilidad

Fiabilidad - Disponibilidad

El atributo de calidad denominado disponibilidad, refiere a la capacidad de que el sistema esté total o parcialmente operativo para que el usuario pueda llevar a cabo las funcionalidades cada vez que lo requiera [24].

La disponibilidad juega un rol importante en la aplicación debido a que esta podrá ser la herramienta de trabajo para algunos de los usuarios. Es por dicha razón, que la no disponibilidad de la aplicación puede significar que el mismo no pueda desempeñar su labor. Para esto el sistema se hospedará en un servidor interno de la empresa. Este servidor, según los datos provistos, estuvo fuera de servicio por cuatro horas en el último año y 6 en el año anterior. Esto nos asegura que el servidor esta funcional más de un 99% del tiempo. Además, se trazó un plan de contingencia en conjunto con el Administrador de redes, tal que se tenga un servidor secundario pronto para ser puesto en funcionamiento en caso de indisponibilidad del principal.

12.4.10.2. Usabilidad

La usabilidad es de los principales requerimientos no funcionales que se deben favorecer en el producto. Dado que los usuarios utilizarán la aplicación estando en la calle, la misma debe poder ser utilizada en esas condiciones sin presentarle un inconveniente al mismo.

Según la definición dada por D. Mordecki [25], "(...) *La usabilidad es la disciplina que tiene como objetivo reducir al mínimo las dificultades de uso inherentes a una herramienta informática, analizando la forma en que los usuarios utilizan las aplicaciones y sitios web con el objetivo de detectar los problemas que se les presentan y proponer alternativas para solucionarlos, de modo que la interacción de dichos usuarios con las aplicaciones y sitios web sea sencilla, agradable y productiva*".

Logrando una buena usabilidad en el sistema se obtienen ciertos beneficios que ayudarán al futuro de la aplicación.

- **Usuarios satisfechos:** Es el resultado de que los usuarios puedan conseguir sus objetivos con el mínimo esfuerzo posible.
- **Usuarios más fieles:** la facilidad de uso provoca que los usuarios deseen volver a utilizar la plataforma y de seguir indagando en sus funcionalidades.
- **Menor costo de soporte:** una aplicación fácil de usar genera menos problemas a los usuarios por lo cual se reduce la necesidad de soporte y ayuda.
- **Menor costo de mantenimiento:** los problemas de usabilidad surgen de quejas de los usuarios, generando modificaciones, por lo cual es mejor realizar las aplicaciones lo más usables posibles desde una primera instancia.

Usabilidad – Facilidad de Operación

Para asegurar la operatividad todas las operaciones se desarrollaron todas las funcionalidades de la aplicación, tal que no tome más de 3 *clicks* el acceder a ninguna de ellas. La implementación de un menú lateral fue una herramienta de gran ayuda para poder cumplir con esto.

Usabilidad – Estética de la interfaz de usuario

Para cumplir con este atributo de usabilidad del sistema se realizaron validaciones y revisiones con expertos para asegurar la misma, mejorando la UI. Estas llevaron a varios cambios en la aplicación. Se pueden ver más detalles de esto en el capítulo de Aseguramiento de Calidad.

Usabilidad – Facilidad de Aprendizaje

Para cumplir con este atributo, no solo se minimizó la cantidad de *clicks* necesarios para acceder a las distintas funcionalidades, sino que también se utilizó una interfaz intuitiva y sencilla de comprender.

12.4.10.3. Eficiencia

La eficiencia es el tiempo requerido del sistema para responder a determinado evento.

Eficiencia – Comportamiento temporal

La aplicación fue desarrollada para poder procesar de forma asincrónica las consultas que manejan mayor cantidad de datos, por ende, las más factibles a demorar más de 5 segundos, como lo es: obtener las paradas, obtener datos de recorridos y obtener próximos arribos a las paradas. De esta forma el usuario puede utilizar la aplicación sin percibir que existen procesos corriendo al mismo momento.

Eficiencia – Capacidad

Para poder asegurar que el sistema soporta 5000 usuarios, se realizaron pruebas de carga con una cantidad determinada de usuarios accediendo de forma concurrente. El equipo se basó en un cálculo fundado en la teoría de M. Baron para poder obtener este valor.

Se definen a continuación las variables necesarias para el cálculo:

Variable	Significado
C	Número de usuarios concurrentes.
n	Número de usuarios totales.
L	Tiempo promedio de uso de la aplicación para todos los usuarios.
T	Intervalo de tiempo durante el cual estamos realizando la medición.

Ilustración 12-61 Significado variables teoría M. Baron

Número de usuarios totales (n): La cantidad de usuarios que se logró alcanzar con el desarrollo fue de 3800, lo cual incluye a todos los inspectores, conductores, guardas, conductores-cobradores y largadores. Como conclusión: **n = 5000 usuarios**.

Tiempo promedio de uso de la aplicación para todos los usuarios (L): Teniendo como base los tiempos que llevo a usuarios algunas de las funcionalidades del sistema (Anexo facilidad de Aprendizaje), se realizó una estimación del promedio de utilización de la aplicación por cada vez que se utiliza la misma, lo cual arrojó como resultado: **L = 9 minutos**.

Intervalo de tiempo durante el cual estamos realizando la medición (T): En promedio la jornada laboral de los empleados es de 8 horas (8 horas), y en general se trabaja de lunes a sábado (6 días). Por lo que, haciendo cuentas, tendríamos: 6 días x 8 horas x 60 minutos = 2880, entonces: **T = 2880 minutos**.

Finalmente calculamos el número de usuarios concurrentes con los cuales deberíamos probar nuestros servicios, esto es hizo siguiendo la presente formula: $C = (n * L) / T$.

$$C = (5000 \text{ usuarios} * 9 \text{ minutos}) / 2880 \text{ minutos} = \underline{16}$$

12.4.10.4. Mantenibilidad

Mantenibilidad - Modificabilidad

La modificabilidad de un sistema apunta al costo e impacto que lleva aplicar cambios en la plataforma. [24]

Uno de los desafíos más grandes para el proyecto fue diseñar una arquitectura que se acople al cliente y pueda ser modificable por ellos. Para lograr esto se realizó una investigación sobre cómo es el desarrollo en .NET en la empresa, qué arquitectura manejan y sus estándares.

En base a los resultados de la investigación, se diseñó una arquitectura que contemple las necesidades del proyecto y el estándar del cliente.

Para lograrlo, se diseñaron capas lógicas con responsabilidades bien asignadas, lo que facilita determinar el impacto al momento de realizar modificaciones para un nuevo negocio, e incorporar nuevos componentes en caso que sea necesario.

Asimismo, cada módulo que representa a usuarios distintos están independizados uno del otro y de esta forma un cambio en la funcionalidad de un módulo no afecta a otro.

Dada la existencia de funcionalidades genéricas útiles para todos los usuarios, existe un módulo general el cual la aplicación siempre accede para obtener estas funcionalidades

El proyecto no contó con un caso en el que 2 usuarios compartan funcionalidades, pero de ser así primero se evalúa la posibilidad de que sea común a todos y de no serlo se deberá crear un módulo específico para esta funcionalidad y de esta forma poder identificarla fácilmente en el sistema. Esto último favorece al atributo de calidad en cuestión, pero afecta la eficiencia. Dada la naturaleza de los requerimientos obtenidos, esta característica es poco común por lo que no tendrá un gran impacto en la eficiencia.

El manejo de error y la seguridad son transversales a todo el sistema por lo que se encuentran en un módulo aparte, accesible por todos.

12.4.10.5. Seguridad

Seguridad - Confidencialidad

Para cumplir con este atributo de calidad, el sistema fue diseñado en función de módulos independientes, los cuales agrupan funcionalidades comunes a distintos roles.

Los roles son los que se usan actualmente en la empresa, con la salvedad de que, para el proyecto realizado, solo se llegó a alcanzar algunos de ellos, pero de todos modos hay que tener en cuenta que el sistema se va a extender a nuevos roles de usuarios en el futuro.

Al momento del inicio de sesión de un usuario, el *backend* verifica el rol del mismo en la base de datos y solamente le ofrece a éste las funcionalidades para las cuales dicho rol está habilitado a usar.

Seguridad – Autenticidad

Para cumplir con este atributo se implementó el uso de sesión de usuario. Para poder garantizar la identidad de quien está operando, se debió asegurar que no pasen largos periodos sin utilizar la aplicación, sin que tenga que re identificarse en el sistema, evitando así el posible uso indebido por parte de un tercero.

Hay 2 motivos por los cuales se le puede pedir a un usuario que se vuelva a identificar:

- Si tuvo un tiempo de inactividad mayor a 30 minutos (sesión expirada por inactividad).
- Si desde que se identificó en el sistema pasaron 8 horas (duración de una jornada laboral), independientemente si utilizo alguna funcionalidad de la aplicación o no, obligando a los usuarios, a identificarse al menos 1 vez por día al comenzar la jornada.

Para la implementación de la sesión de usuario, se utilizaron *tokens* autogenerados que identifican a un usuario específico con una sesión.

Seguridad – Integridad

Para cumplir con este atributo se tomaron dos medidas distintas, en primer lugar, se decidió cifrar la contraseña de los usuarios tanto en la Base de datos como del lado de la aplicación móvil al momento de ser digitada, viajando hasta el *backend* de forma

encriptada. Para el encriptado se utilizó AES como algoritmo de encriptación simétrica, utilizando una única clave de encriptación de 128 bits de tamaño.

Por otro lado, se utilizó un canal de comunicación seguro para todas las comunicaciones realizadas entre la aplicación móvil y el *backend*, utilizando TLS/SSL y un certificado de seguridad. De esta forma la información viaja encriptada y no puede ser interceptada por terceros.

Durante la implementación del encriptado del canal, se investigó sobre distintos tipos de certificados, descubriendo que los que se podían crear por defecto utilizando la herramienta de IIS, son del tipo SHA-1, función hash que ya fue quebrada previamente [42] y además va a dejar considerada segura por Google en los próximos meses [29]. Investigando, se logró crear e instalar en el IIS un certificado más seguro del tipo SHA-2, mismo tipo de certificado utilizado por entidades bancarias, por ejemplo. El cual aún no fue quebrado y brindara mayor seguridad a las comunicaciones del sistema.

12.4.11. Avances de Investigación

Se realizaron investigaciones sobre posibles soluciones a los problemas que puede presentar la aplicación. Se listan a continuación las distintas pruebas realizadas.

PRUEBA GPS

Dada la posición actual del usuario, permite determinar qué parada se encuentra más cercano a él, y obtener información sobre las líneas que pasan en dicha parada. A su vez, al seleccionar una línea, mostrar los horarios de la misma.

Nota: Las paradas, líneas y horarios son a modo de ejemplo, y se usaron de prueba dentro del complejo de C.U.T.C.S.A. en dos puntos separados en una habitación, donde el *tester* se acercaba a una posición o a otra y veía cómo cambiaba la parada más cercana.

PRUEBA NFC

Pasar una tarjeta *mifare* simulando que tiene cargada una lista de tarjetas, almacenarlas en el dispositivo, y luego verificar con otras tarjetas si las mismas (sus identificadores) pertenecen a la lista o no.

PRUEBA BD

Obtenida de la guía de Android sobre Bases de Datos de la materia Desarrollo de Aplicaciones Móviles dictada en el primer semestre del 2014 en la Universidad ORT Uruguay, permite guardar nombres en la memoria del dispositivo móvil, y mostrarlos en pantalla en una caja de texto. También tiene la opción de mostrar los registros en una lista.

PRUEBA LECTURA DE CÓDIGOS QR Y BARRAS

Usando la aplicación *Barcode Scanner* obtenida del Google Play, permite obtener información de los datos almacenados en un código QR o de barras.

PRUEBA WS REST

Acceder a un servicio web de Mercado Libre que obtiene información del país, estado, ciudad y código postal ingresando el código del país y un código postal, a través de REST. Ejemplo:

- Código de país: AR
- Código postal: 5000

PRUEBA WS SOAP

Acceder a un servicio web que permite convertir grados Celsius a Fahrenheit, a través de SOAP. Ejemplo: Valor Celsius: 40

PRUEBA OSM

Mostrar un mapa de *Open Street Maps* y simular las mismas funcionalidades que podemos hacer con *Google Maps*:

- Mantener apretado sobre el mapa y colocar un marcador
- Escribir una posición y colocar un marcador sobre el mapa en dicha posición
- Seleccionar un marcador y mostrar información
- Zoom in y Zoom out.
- Obtener la ubicación del usuario y mostrarla en el mapa.

PRUEBA AUTOCOMPLETE

Buscar una dirección y obtener información sobre ella utilizando el Google Places API: nombre, id, dirección, teléfono, *website* y posición.

Se puede buscar una calle (ej. Avenida Brasil, Montevideo, Departamento de Montevideo, Uruguay), o también con número de puerta:

- Avenida Brasil 1950, Montevideo, Departamento de Montevideo, Uruguay
- Avenida Brasil, Montevideo, Departamento de Montevideo, Uruguay, 1950
- Avenida Brasil, Montevideo, Departamento de Montevideo, Uruguay 1950
- 1950 Avenida Brasil, Montevideo, Departamento de Montevideo, Uruguay

PRUEBA MAPA AUTOCOMPLETE

Buscar una dirección o una posición geográfica y posicionarla con un marcador sobre el mapa. Ejemplo:

- Avenida Brasil, Montevideo, Departamento de Montevideo, Uruguay 1950
- -34.910608, -56.152309

También permite posicionar la ubicación del móvil (obtenida por GPS) sobre el mapa.

PRUEBA FIRMAS IBUS

Obtener información desde los *web services* de Movistar utilizando JSON y REST. Funcionalidades:

- Paradas: Obtener información sobre paradas y mostrarlas en el mapa. Al seleccionar una parada sobre el mapa, permite ver su información (número de parada, dirección y líneas que pasan por ella).
- Mi posición: Posicionar la ubicación del móvil (obtenida por GPS) sobre el mapa, presionando el botón “o”.
- Buses Parada: Dado un número de parada (ej. 2683) y/o un número de línea (ej. 156), muestra información sobre la parada (número y dirección), información sobre la línea (número y destino), y en cuánto tiempo llega dicha línea a la parada.
 - Si se coloca sólo la línea, busca la parada más cercana al usuario y obtiene las predicciones para dicha línea;
 - Si coloca sólo número de parada, muestra las predicciones de las próximas líneas que llegan a la parada;

- Si coloca número de parada y número de línea, si la línea pasa por la parada, muestra la predicción solamente de la línea indicada.

- Posición Bus: Obtener la posición geográfica del coche indicado (puede no retornar datos en caso que el coche no tenga el GPS activado).
- Buscar Ruta: Colocando una posición origen y posición destino (latitud y longitud), y seleccionar la ruta más rápida o más económica, muestra información sobre dónde ir, qué parada esperar una línea (junto con el tiempo de espera y tiempo de viaje), dónde bajarse, etc. hasta llegar a destino. Ejemplo: Tipo de ruta: el más rápido
- Boleterías: Obtener información sobre locales de ventas (nombre, dirección, barrio, teléfono, horarios) y mostrarlos en el mapa.

12.5. Anexo 5 - Plan de Proyecto

12.5.1. Resumen ejecutivo

El desarrollo de un proyecto de fin de carrera trae varios desafíos a la hora de la gestión. Por dicha razón es que este documento tiene como objetivo analizar los factores que podrían influir en el proyecto, tanto externos como internos al equipo. Se analizarán las suposiciones, restricciones, y riesgos para realizar estimaciones y así generar el cronograma a seguir hasta la entrega final del proyecto.

12.5.2. Objetivos del Producto

Crear una aplicación móvil que ayude y apoye la gestión, comunicación y trabajo que realizan los distintos funcionarios en su día a día, en todas las áreas de la empresa.

Permitir que la aplicación dé soporte, agilice y mejore la organización de las tareas, aumente la eficiencia y eficacia del funcionario y mejore o complemente las comunicaciones entre empleados, departamentos y áreas.

De esta forma, el empleado podrá tener de forma más accesible, rápida y sencilla la información que utiliza durante su jornada laboral, aprovechando las ventajas que ofrecen las características de un dispositivo móvil.

12.5.3. Gestión del alcance

El proyecto se dividió en tres ciclos a ser desarrollados, en cada uno realizándose un conjunto de módulos y funcionalidades.

Antes de comenzar, se realizó un análisis de las áreas y departamentos más representativos de la empresa, con el fin de comprender mejor el contexto y relevar posibles requerimientos a desarrollar. Luego, se iniciaron las evoluciones.

La primera evolución consistió en analizar y armar la base de la aplicación móvil y plataforma web que consumirán los distintos módulos que se realizarán posteriormente para el sistema. Aquí se enfocó en el desarrollo de una nueva base de datos, y el desarrollo del código base de la aplicación del lado del servidor, modificable y extensible, donde se colocaron los servicios web a ser consumidos por la aplicación móvil, y que consumirá los datos de la nueva base de datos. También el desarrollo del código base, de la aplicación móvil, también modificable y extensible, para poder soportar el agregado de nuevos módulos con el menor impacto y retrabajo posible. A su vez, se desarrollaron algunas funcionalidades del módulo común.

De forma concomitante, se comenzó con el desarrollo de la segunda evolución, correspondiente a las funcionalidades para los inspectores.

Finalmente, luego de finalizados las evoluciones anteriores, se desarrolló la tercera evolución, en la cual pertenecen algunos requerimientos para los módulos para largadores, guardas, conductores, conductores cobradores, y algunas funcionalidades comunes más.

Se analizaron los posibles entregables que formarán parte del proyecto, concluyendo en la tabla que se encuentra en el [Capítulo de Gestión de comunicaciones](#).

Quedó fuera del alcance el desarrollo del componente “Intermediador”, que hace de conexión entre la base de datos de la empresa y la base de datos de la que la aplicación móvil tomará información. La misma quedará como responsabilidad de la empresa. Las funcionalidades se probaron, agregando ejemplos de datos reales en la base de datos del proyecto.

12.5.4. Alcance del producto

El producto fue compuesto, por un lado, por una aplicación móvil separada por módulos, cada uno ofreciendo funcionalidad para cada área o sector de la empresa y, por otro lado, por el *backend* que contó con el desarrollo de los servicios web que consumirá la aplicación móvil, así como el desarrollo y gestión de la base de datos que ofrecerá la información necesaria.

A través del análisis de los requerimientos funcionales más importantes del proyecto, que puede apreciarse de forma detallada en el [anexo Análisis de posibles requerimientos a desarrollar](#), y la estimación realizada en el capítulo de [Capítulo de Estimaciones](#), el alcance del producto fue definido de la siguiente manera:

Opciones generales para los funcionarios:

- Inicio y cierre de sesión.
- Menú con opciones según su rol del usuario en la empresa.
- Reportes de error (envía una notificación a los desarrolladores sobre correcciones o mejoras dentro de la aplicación).
- Gestión de notificaciones.
- Visualización de noticias.

Opciones para inspectores:

- Ver información de paradas (posición geográfica, líneas que pasan por ella).
- Ver recorridos y sus paradas en el mapa.
- Ver próximos arribos de coches (de cada línea) a una parada seleccionada.
- Ver la posición geográfica de los coches de líneas en el mapa, e información de su servicio.
- Obtener cómo llegar desde un origen a un destino usando líneas de C.U.T.C.S.A.
- Ver la información y posición geográfica de distintos puntos de interés en el mapa.
- Controlar el servicio de un coche (incluyendo el control de boletos expendidos).
- Ver el horario planificado de recorridos.

Opciones para conductores, guardas y conductores cobradores:

- Consulta de servicios asignados.

Opciones para largadores:

- Modificación, alta y baja de servicios y viajes a realizar.

12.5.5. Estimaciones

En una primera instancia, las estimaciones se realizaron por analogía, *bottom up* y juicio experto, en función de experiencias pasadas en obligatorios y proyectos de desarrollo similares, teniendo en cuenta el análisis de los registros de horas realizados hasta el momento.

A su vez, existen restricciones de tiempo que delimitaron la fecha final del proyecto, los cuales se tuvieron en cuenta para la organización de las etapas que aparecen en el cronograma.

También se tuvieron en cuenta los riesgos detectados en el proyecto para aplicar planes de entrenamiento y extensión de las duraciones de algunas tareas. En particular, el aprendizaje para desarrollar en Android, dado que el producto se basa en ello, el cual se explica más profundamente en el [Capítulo de Gestión de riesgos](#).

Luego de haber finalizado con la Especificación de Requerimientos de las primeras dos evoluciones, y gracias al análisis de puntos de función que se realizó para sus módulos, se pudo mejorar la precisión de las estimaciones para estos dos puntos del ciclo de vida (Especificación de requerimientos y Diseño) para la tercera evolución.

A su vez, luego de haber finalizado un conjunto de requerimientos del módulo general (*Login*, Menú, y Reportes de error), se pudo utilizar el registro de horas dedicadas al

Desarrollo y Prueba en la estimación de los requerimientos del resto de los requerimientos.

Al finalizar la actividad de relevamiento para la tercera evolución, junto con un nuevo análisis de puntos de función, se logró estimar el tiempo a dedicar para el desarrollo y *testing* de la misma.

12.5.6. Cronograma de trabajo

A continuación, se muestra el cronograma de las tareas que el equipo tiene planificado para realizar en el proyecto. Para la organización de las tareas, se utilizó las instancias de las evaluaciones en la universidad: Revisiones y entrega final. Al final de cada una de ellas, se estipuló la finalización de cada etapa.

Nombre de la etapa	Duración	Comienzo	Fin	Descripción
Etapa inicial	51 días	mar 28/04/15	mié 17/06/15	<p>Desarrollar la descripción inicial de la empresa para entender mejor el problema y relevar posibles requerimientos del sistema que ayudarán a la decisión de los módulos a desarrollar.</p> <p>A su vez, corresponde a la diagramación de planes que apoyarán el desarrollo del proyecto de la carrera.</p> <p>Concluir con la primera revisión, a través de la realización de las presentaciones correspondientes a lo realizado para el proyecto hasta el momento.</p>
Segunda Etapa	69 días	jue 18/06/15	lun 03/09/15	<p>Comenzar con las correcciones que surgieron de la primera Revisión.</p> <p>En paralelo, realizar actividades relacionadas al aprendizaje en desarrollo para Android, con el fin de mitigar el riesgo de falta de conocimiento del mismo.</p> <p>Luego, comenzar con la primera y segunda evolución. Sus</p>

				<p>actividades se extenderán hasta la Etapa Final.</p> <p>Al final de la etapa, realizar la segunda revisión, a través de la realización de las presentaciones correspondientes a lo realizado para el proyecto hasta el momento.</p>
Tercera Etapa	56 días	mar 04/09/15	dom 08/11/15	<p>Comenzar con las correcciones que surgieron de la segunda Revisión.</p> <p>Continuar con las primeras dos evoluciones.</p> <p>Concluir con la tercera revisión, a través de la realización de las presentaciones correspondientes a lo realizado para el proyecto hasta el momento, y luego dedicar un tiempo para realizar correcciones.</p>
Etapa Final	133 días	lun 09/11/16	lun 03/03/16	<p>Comenzar con las correcciones que surgieron de la segunda Revisión.</p> <p>Finalizar las primeras dos evoluciones, y comenzar con la tercera.</p> <p>Realización de los distintos documentos necesarios para la entrega en la facultad y la empresa como conclusión del proyecto de fin de carrera.</p>

Tabla 12-22 Descripción general de tareas a realizar en cada etapa

Para más detalles del cronograma, ver en el [anexo del Cronograma del proyecto](#).

A continuación, se puede apreciar la estructuración de las tareas a planificadas para desarrollarse durante el proyecto.

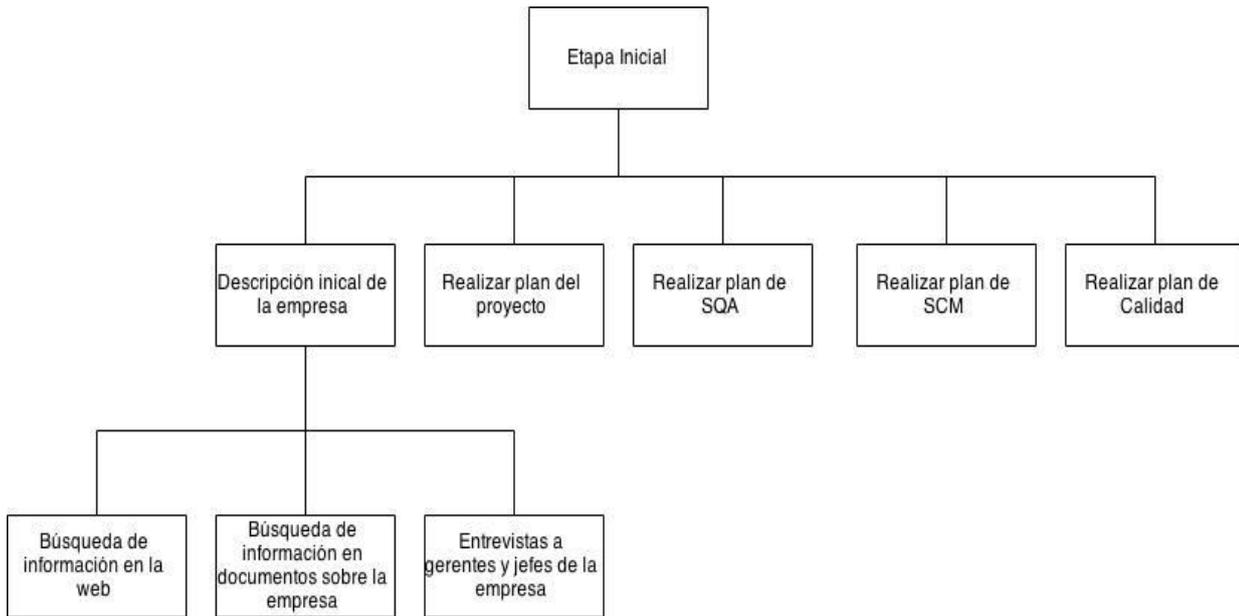


Ilustración 12-62 Tareas a realizar antes de las evoluciones

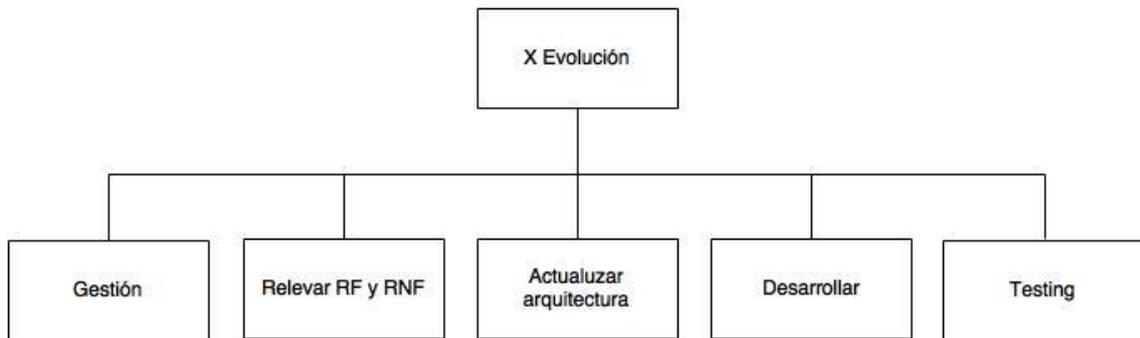


Ilustración 12-63 Tareas a realizar en cada evolución



Ilustración 12-64 Tareas a realizar luego de las evoluciones

12.5.7. Análisis de la evolución del cronograma

Aquí se explican cómo se fue desarrollando las actividades definidas en el cronograma. La siguiente ilustración muestra, por cada actividad del proyecto, cuál fue su planificación (verde, barra superior), y la ejecución real (azul, barra inferior).

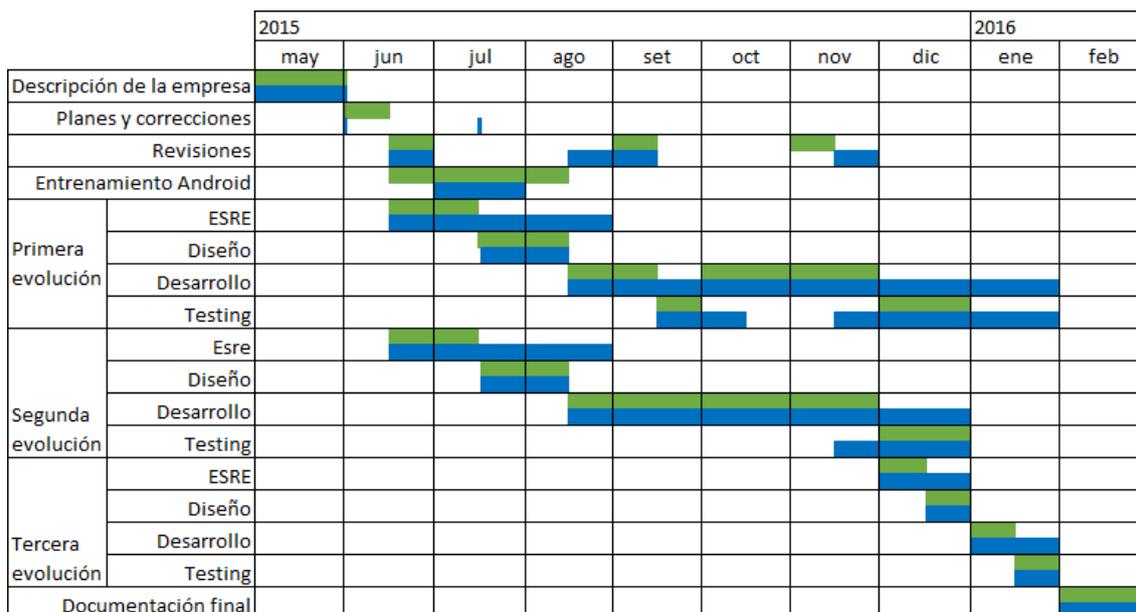


Ilustración 12-65 Cronograma simplificado del proyecto, conteniendo la planificación (verde) y ejecución (azul) de tareas

Las actividades de la Etapa Inicial se realizaron tal como se estimó, a excepción de los planes SCM, SQA y calidad, que se retrasaron por las evaluaciones que se debieron realizar en ese momento.

Con respecto al entrenamiento sobre Android, se comenzó más tarde por la necesidad de comenzar con la actividad de especificación de requerimientos de la primera evolución. A su vez, se debió suspender por los retrasos que se estaban registrando en las demás actividades, concluyendo en la decisión de que el integrante que tenía experiencia en Android, realizara el desarrollo completo de la aplicación.

La segunda revisión se comenzó antes para ir generando las presentaciones de forma anticipada, teniendo la revisión a principios de setiembre. La tercera revisión se comenzó más tarde porque la asignación por parte de la facultad fue en una fecha más tardía a la planificación.

En las tres evoluciones, hubo un retraso en la especificación de requerimientos, tal como se comentó anteriormente, a raíz de las evaluaciones que tenían dos de los integrantes del proyecto. En las primeras dos evoluciones, el retraso no fue por el relevamiento en sí, sino por la generación del documento, por lo que no afectó la etapa de diseño. Para la tercera evolución, el diseño se retrasó debido a que en este caso sí dependía de la especificación.

El desarrollo y prueba de la primera evolución debió extenderse hasta fines de enero, por problemas en la solución sobre la seguridad del sistema, dado que ninguno de los integrantes tenía fuertes conocimientos sobre el mismo. El resto del desarrollo y prueba se realizó según lo planificado.

Para todas las evoluciones, el desarrollo se extendió incluso durante la etapa de *testing*, por retrabajo y correcciones.

En el caso de la primera evolución, el *testing* consumió más días de lo planificado por la falta de métricas para realizar una estimación precisa.

En la segunda evolución, el *testing* comenzó de forma temprana, cuando se finalizó el desarrollo del *frontend*, y de esta forma, comenzar con las *testing* del *backend*.

Para una descripción más detallada sobre el esfuerzo dedicado y comentarios al respecto, ver el [Capítulo de Gestión del Esfuerzo](#).

12.5.8. Gestión de riesgos

La siguiente tabla muestra los riesgos más importantes a los que se le realizaron seguimiento durante todo el proyecto, basados en su probabilidad e impacto.

Id	Riesgo	Plan de Respuesta	Plan de Contingencia	Estrategia del riesgo
1	Varios integrantes del equipo no tienen conocimiento sobre desarrollo para Android, lo cual puede retrasar la etapa de diseño y desarrollo.	(1) Realizar un plan para aumentar la experiencia del desarrollo. (2) Cuando exista capacidad ociosa, aprovechar a hacer pruebas y "agarrarle la mano" al desarrollo.	(1) Invertir más tiempo en desarrollo. (2) Desarrollar un plan de entrenamiento	Mitigar

		(3) Proporcionar guías y ayudas para desarrollo.		
2	No existe un conjunto de requerimientos iniciales o básicos para el desarrollo del sistema (aparte del sistema para los inspectores), lo cual puede generar requerimientos inestables, que si se modifican durante el desarrollo (o luego) traerá retrabajo.	(1) Análisis del dominio y contexto. (2) Separar el producto en módulos lo más independientes posible para (3) Análisis exhaustivo en cada módulo para obtener requerimientos (preguntar a varios funcionarios del departamento).	Invertir más tiempo en desarrollo.	Mitigar
3	El sistema es para todas las áreas de C.U.T.C.S.A. (empresa muy grande), que tendrá muchos módulos y comunicaciones. Dicho problema puede hacer que se dedique mucho tiempo sobre diseño y arquitectura que puede retrasar el desarrollo.	(1) Separación del sistema por módulos. (2) Realizar un relevamiento de RF y RNF genéricos del proyecto y del sistema en su totalidad.	Invertir más tiempo en desarrollo.	Mitigar
4	El equipo tiene poco conocimiento sobre desarrollo de seguridad en un sistema, y su mala implementación puede traer insatisfacción del cliente. Hay que invertir tiempo sobre e arquitectura	(1) Basarnos en sistemas de seguridad empleadas por la empresa. (2) Ayuda de expertos (en la empresa, en la facultad, etc.).	Invertir más tiempo en análisis, investigación y desarrollo.	Mitigar

	investigación que puede retrasar el desarrollo.	(3) Buscar información.		
5	Existe la probabilidad de que varios interesados del proyecto de quienes dependemos para realizar decisiones se no se encuentren disponibles por un período de tiempo y no nos den una respuesta a tiempo.	Especificar fechas límites, avisando al interesado.	En caso de superar dicha ficha sin obtener respuesta, la decisión será realizada por el equipo del proyecto.	Eliminar
6	Falla en la estimación para el cronograma	Realizar un pequeño ciclo de un requerimiento en particular para tomar métricas y mejorar la estimación.	(1) Aumentar el esfuerzo para cumplir con la estimación. (2) Volver a estimar en función a datos anteriores	Mitigar
7	Existe una dependencia sobre servicios que no podemos realizar nosotros por falta de datos, y que deben ser desarrollados por la empresa Handsoft. Dichos servicios son saber la posición de un coche dado su número; y saber cuándo demoraría un ómnibus de cierto recorrido trasladarse desde un punto inicial a un punto final. A su vez, se debe esperar por un prototipo de la tarjeta que utilizarán los inspectores para culminar	Esperar por la confirmación de las fechas en que tendremos disponibles estos requerimientos, y realizar replanificación. Mientras no se nos dice la fecha, se realizará con información que se tenga actualmente (por ejemplo, posición planificada donde debería estar el coche según la planificación) y simuladores de dicha información.	-	Mitigar

	dicho desarrollo y realizar las pruebas pertinentes.			
--	--	--	--	--

Tabla 12-23 Lista de riesgos tenidos en cuenta para el proyecto

En la tabla anterior, la columna plan de respuesta se refiere a acciones para que el riesgo no se convierta en problema, la de plan de contingencia se refiere a acciones en caso que el riesgo se haya convertido en problema. Para visualizar el resto de los riesgos analizados, pero a los que no se les hizo seguimiento, ver el anexo [Análisis de otros riesgos](#).

A continuación, se representa el seguimiento de los riesgos mencionados, y su evolución a lo largo del proyecto. Dicho análisis se realiza al comienzo de cada etapa del proyecto.

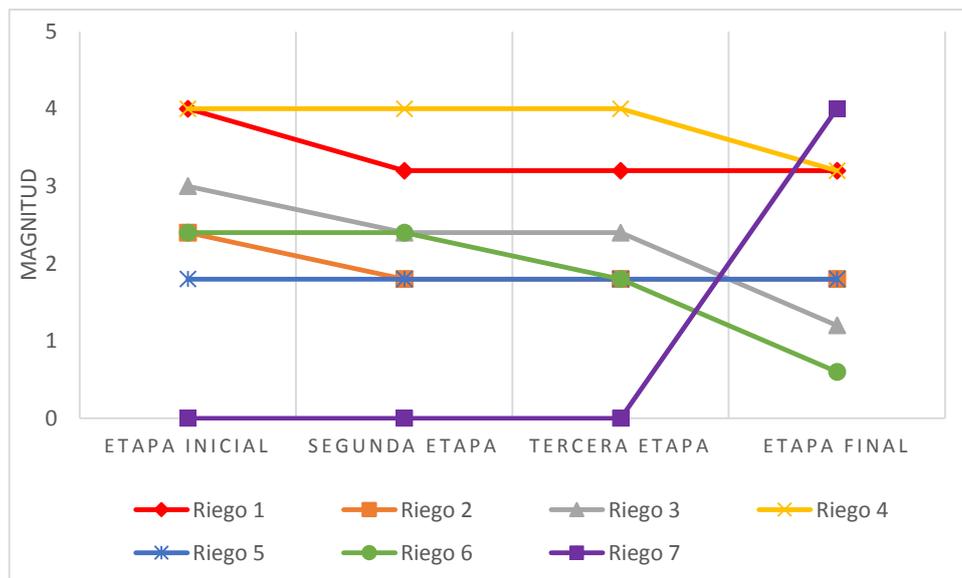


Gráfico 12-14 Riesgos a los que se hizo su seguimiento y su magnitud

Para el cálculo de la magnitud, se realizó la siguiente fórmula:

$$\text{Impacto} * \text{Probabilidad de ocurrencia} = \text{Magnitud}$$

Ecuación 1 Calculo de magnitud

Impacto	
Valor	Descripción
0	Ninguno
1	Marginal
2	Poco importante
3	Importante (puede retrasar un poco el proyecto)
4	Muy Importante (puede retrasar mucho el proyecto)
5	Crítica (puede detener el proyecto)
6	Catastrófica (fracaso del proyecto)

Tabla 12-24 Valores del impacto tenidos en cuenta para el cálculo de la magnitud

Probabilidad de ocurrencia	
Valor	Descripción
0	Improbable
0.2	Poco probable
0.4	Probable
0.6	Muy probable
0.8	Altamente probable
1	Se convierte en problema

Tabla 12-25 Valores de la probabilidad de ocurrencia tenido en cuenta para el cálculo de la magnitud

Para mitigar el Riesgo 1 (falta de conocimiento en desarrollo Android), se realizó un plan de entrenamiento para aprender a desarrollar en Android, el cual tuvo un período de aproximadamente dos meses. El mismo pretendía ayudar a aprender los conceptos básicos para el desarrollo Android, involucrando interfaz gráfica, Intentos (uso de cámara), almacenamiento en base de datos, acceso a servicios web mediante REST, tareas concurrentes, y mapa.

Pero dicho entrenamiento no pudo concluirse dado que se debieron dedicar horas en tareas más importantes, y dicho riesgo no se pudo mitigar. Por dicha razón, el desarrollo de la aplicación móvil lo desarrolló el integrante experto en desarrollo Android, mientras que el resto de los integrantes dedicaron su tiempo al desarrollo del *backend*.

Para mitigar el Riesgo 2 (falta de requerimientos iniciales para el proyecto), se decidió invertir tiempo en una etapa de relevamiento inicial, para entender cómo funciona la empresa y obtener posibles requerimientos a incluir en el sistema. A su vez, se puso énfasis en la etapa de especificación de requerimientos, teniendo varias reuniones para obtener requerimientos, validarlos, ir mostrando bocetos y cómo fue avanzando el diseño.

Para mitigar el Riesgo 3 (dedicación en diseño y arquitectura), se realizó un ciclo de ingeniería inicial, al que se llamó Módulo Común, que permitió definir todos los elementos que serían comunes para los distintos módulos a desarrollar y así poder realizar componentes reutilizables para disminuir el tiempo de desarrollo.

Para mitigar el Riesgo 4 (falta de conocimiento en seguridad informática), uno de los integrantes se enfocó en la solución de estos problemas en la etapa de desarrollo, apoyándose en expertos y profesores de la Universidad ORT para encontrar distintas soluciones al problema.

Para mitigar el Riesgo 5 (falta de disponibilidad de interesados en el proyecto), siempre que existió la posibilidad, se anticipó y agendó reuniones antes que se aproximaran las distintas fechas especiales (feriados, fin de semanas largos, licencias, vacaciones).

El Riesgo 6 (falla en la estimación) se pudo mitigar gracias al análisis de puntos de función y registro de horas, realizadas en la Tercera Etapa, que permitieron estimar de mucha mejor manera las distintas tareas del proyecto.

Hasta la Tercera Etapa, no se había tenido en cuenta el Riesgo 7 (Dependencia de sistemas externos a la empresa), dado que, por decisión del cliente, se debía realizar el sistema sin tener que depender de sistemas externos a la empresa. Se consultó tomar código de dichos sistemas (dado que C.U.T.C.S.A. es propietaria) y así culminar con toda la funcionalidad necesaria. Pero a la hora de intentar reproducir los algoritmos necesarios, en dos de las funcionalidades (obtener posición geográfica del ómnibus, y cuántos minutos demora un ómnibus en recorrer desde un punto inicial a un punto final), se detectó que no se podían culminar por falta de referencias entre un número de coche y su posición (en el caso de la posición del ómnibus), y la desprolijidad del proyecto que contenía el algoritmo del cálculo del tiempo que demoraba el ómnibus en ir desde una posición a otra (la cual no se pudo entender el código). Se consultó para tener servicios especiales que ofrecieran al equipo del proyecto las soluciones a los datos faltante, lo cual se les entregó en tiempo y forma para concluir con el desarrollo.

A su vez, para el caso de las tarjetas puentes para el control de servicios, hay datos faltantes en la plantilla entregada por la I.M.M., y que son importantes para dicho control. Junto con el *Sponsor*, se decidió dejar en suspenso dicha funcionalidad hasta que se llegue a un acuerdo entre la empresa y la Intendencia.

12.5.9. Gestión del Esfuerzo

En este capítulo se describe una evaluación y análisis del esfuerzo para cada cierto período de tiempo, realizados al final de cada revisión.

12.5.9.1. Etapa inicial

En la etapa inicial, se comenzó utilizando una planilla en Google Drive para el registro de tareas y horas invertidas por cada miembro del equipo, pero las horas no fueron registradas por varios de los miembros del equipo, por lo que dificultó y limitó el análisis del esfuerzo. De todas formas, por evaluaciones que se realizaron a los integrantes del equipo, se determinó que el esfuerzo dedicado para esta primera etapa fue baja a raíz de materias y evaluaciones correspondientes al primer semestre.

A su vez, notamos que se atrasaron las actividades de realización y corrección de los planes. Para el caso del Plan del Proyecto, el retraso fue debido a que se atrasó la fecha para la reunión prevista con el Tutor de gestión del proyecto por dos semanas.

Dichos planes deberán finalizarse en la segunda etapa.

A continuación, se muestra el esfuerzo registrado por uno de los integrantes del equipo, el cual comprende las fechas entre el 27 de abril del 2015 y el 21 de junio del mismo año (8 semanas).

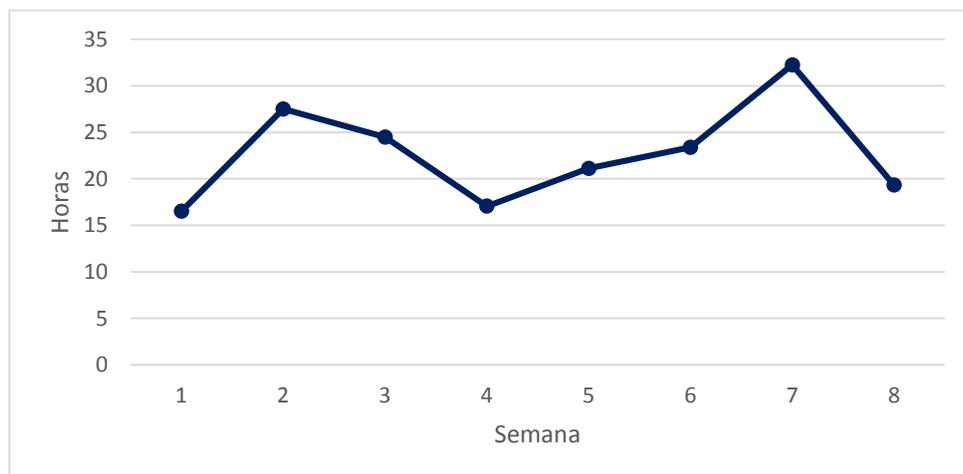


Gráfico 12-15 Esfuerzo realizado por uno de los integrantes durante la primera etapa del proyecto

Como podemos ver en la gráfica anterior, hubo un pico grande en la semana 2 donde se realizaron las entrevistas a distintos responsables de la empresa, y en la semana 7 previa

a la primera revisión, en la cual se debió armar la presentación (diapositivas) para ser utilizada en la misma.

El promedio de dicho período para el integrante fue de 23 horas semanales, sumando un total de 182 horas dedicadas en el mismo.

12.5.9.2. Segunda Etapa

Comprende desde el 22 de junio del 2015 al 06 de setiembre del mismo año (11 semanas).

Con respecto a la planificación de las tareas, se registró un retraso en la creación de la especificación de requerimientos y diseño, lo cual llevó al retraso de actividades posteriores: el comienzo del desarrollo. La razón por la cual se retrasaron dichas tareas fue por la falta de dedicación al proyecto por parte de los integrantes del equipo a raíz de evaluaciones dentro de la facultad. Las dedicaciones a dichas evaluaciones fueron mayores a las estimadas.

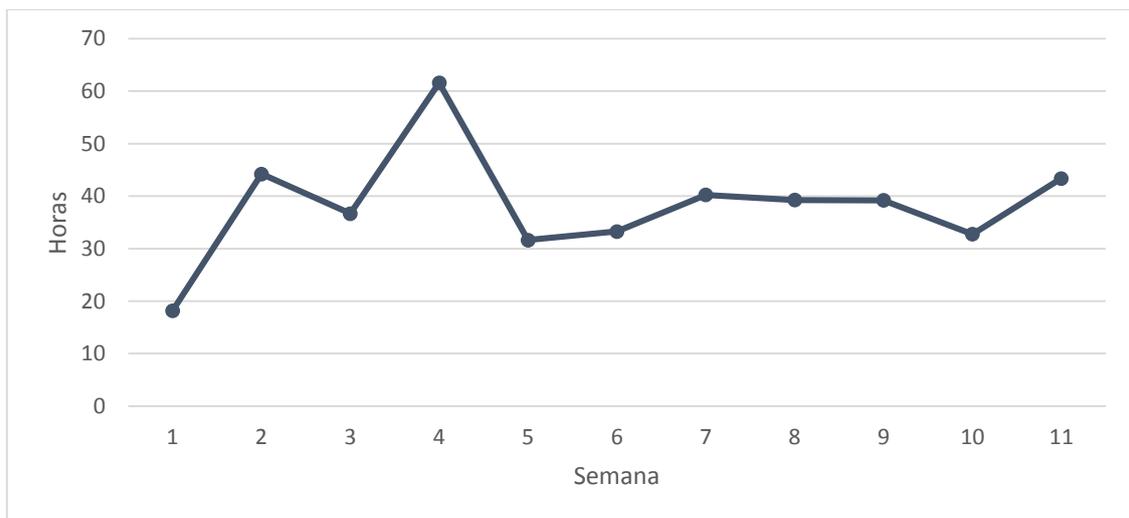


Gráfico 12-16 Esfuerzo realizado por el equipo para la segunda etapa

En la semana 1, Andrés y Pedro no tienen registro de horas. El registro lo comenzaron a realizar a partir de la semana 2 de este período.

Como se puede apreciar en la imagen, existió un pico en el registro de esfuerzo en la semana 4 a raíz de las actividades realizadas para el relevamiento de requerimientos para el módulo de inspectores.

Por parte de Pedro, existió un pico de esfuerzo en la semana 7 dada la dedicación sobre el documento de arquitectura común para todos los módulos y de los inspectores.

La cantidad de horas recomendado por la facultad para la dedicación del proyecto es de entre 15 y 20 horas semanales por integrante, es decir, entre 45 y 60 horas recomendadas para la dedicación promedio de todo el equipo. El equipo registró un promedio de 38 horas semanales.

De esta forma, el total de horas recomendadas para la realización del equipo en este período es de entre 495 y 660 horas, donde el equipo dedicó 420 horas.

En vista a este análisis, se buscó la forma de corregir el problema.

12.5.9.3. Tercera Etapa

Comprende del 07 de setiembre del 2015 al 8 de noviembre del mismo año.

En esta etapa se continuó con el desarrollo y las pruebas, tal como se había planificado.

No se realizó la presentación de la tercera revisión en esta etapa, debido a que se asignó la fecha para el 30 de noviembre del 2015 por la facultad. En respuesta a esto, el tiempo a dedicar para dicha presentación se trasladó para unas semanas más adelante, y se adelantó el desarrollo.

En esta etapa se realizó la reestimación del cronograma, gracias a los datos obtenidos en el desarrollo y *testing* de la primera evolución. Esto, junto con el análisis de puntos de función para cada uno de los requerimientos relevados, permitió afinar la estimación del resto del desarrollo y prueba.

Se debe destacar que en esta etapa se aumentó considerablemente la dedicación del proyecto, resolviendo así el problema de no estar alcanzando con la dedicación mínima propuesta.

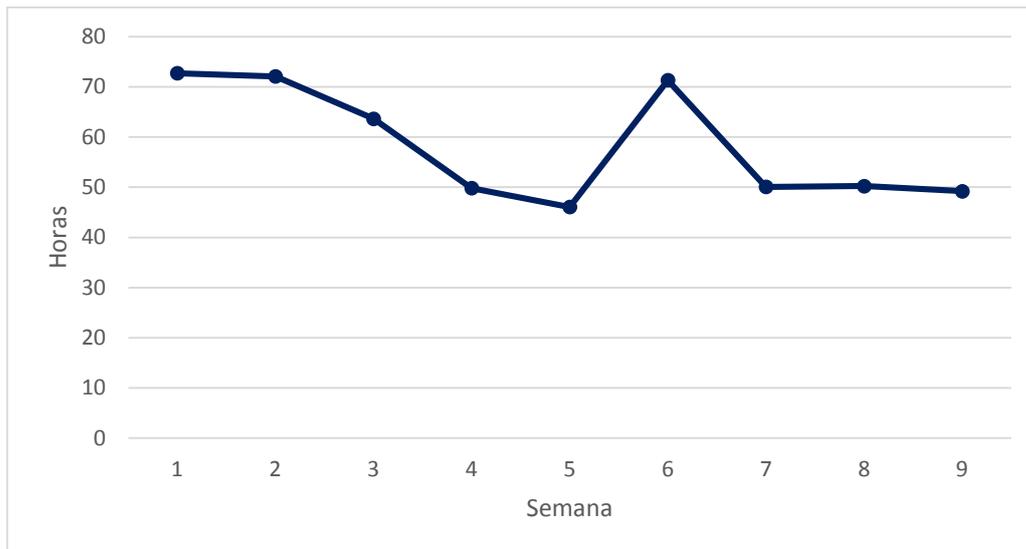


Gráfico 12-17 Esfuerzo realizado por el equipo para la tercera etapa

Tal como se puede apreciar en la gráfica anterior, Diego y Pedro tuvieron un pico importante en las semanas 1 y 2 de este período, en el que se dedicó en gran parte a la finalización del desarrollo para que de esa forma no se retrasara según lo planificado, y así empezar en la semana 3 con las pruebas.

Luego, el resto de la dedicación fue bastante estable, en el que se dedicó al desarrollo y pruebas, con un particular pico en la semana 6 por parte de Pedro para corregir la falta de dedicación en la semana 5.

En conclusión, el equipo tuvo una dedicación semanal promedio de 58 horas, y un total de 525 en esta etapa, permaneciendo de esta manera en el rango semanal comprometido, y de las 405 horas mínimas para el período.

12.5.9.4. Etapa final

Comprende del 9 de noviembre del 2015 al 3 de marzo del 2016, aunque el análisis de horas se realizó hasta el 31 de enero, es decir, hasta la finalización de la tercera evolución.

En esta etapa se completaron las primeras dos evoluciones, y se realizó la tercera evolución de forma completa, tal como se había planificado.

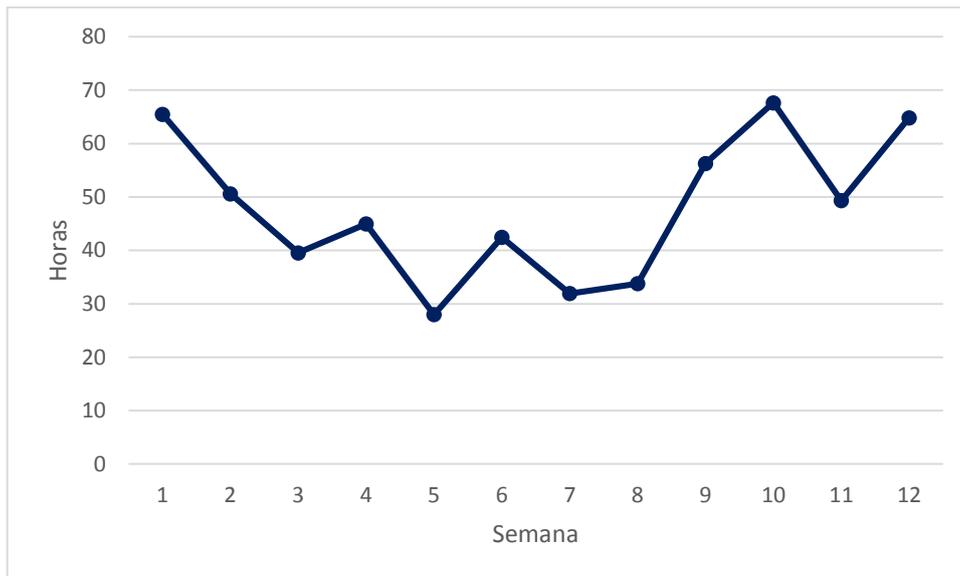


Gráfico 12-18 Esfuerzo realizado por el equipo para la etapa final

Tal como se aprecia en la gráfica anterior, la dedicación fue disminuyendo gradualmente a raíz de las evaluaciones que los integrantes tuvieron en la facultad. Entre las semanas 9 y 12 se volvió a aumentar la dedicación para culminar el desarrollo y pruebas de la tercera evolución, teniendo un valle en la semana 11 por las fiestas.

Para este período, se dedicó un total de 575 horas, con un promedio de 48 horas semanales, superando así las 540 horas totales y 45 horas promedio por semana comprometidas.

12.5.9.5. Conclusión del esfuerzo

A pesar de comenzar con una dedicación menor a la comprometida por el proyecto, de forma global se logró cumplir con el objetivo de superar las 45 horas promedio semanales dedicadas por el equipo, alcanzando las 47 horas en función de los datos reales registrados.

12.5.10. Organización de partes interesadas

Para nuestro proyecto, representamos a los *stakeholders* de la siguiente manera:

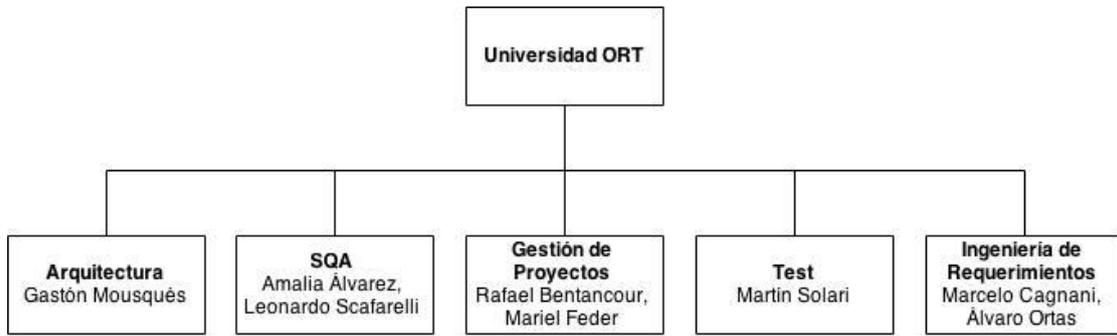


Ilustración 12-66 Stakeholders dentro de la Universidad ORT

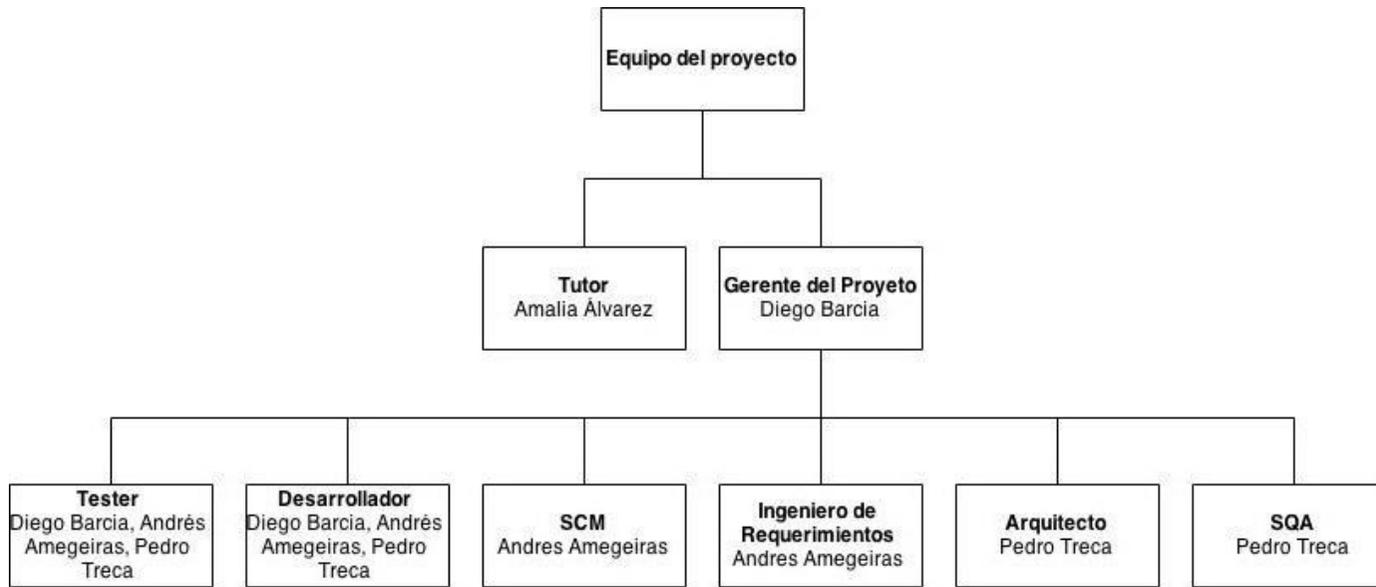


Ilustración 12-67 Stakeholders dentro del equipo del proyecto

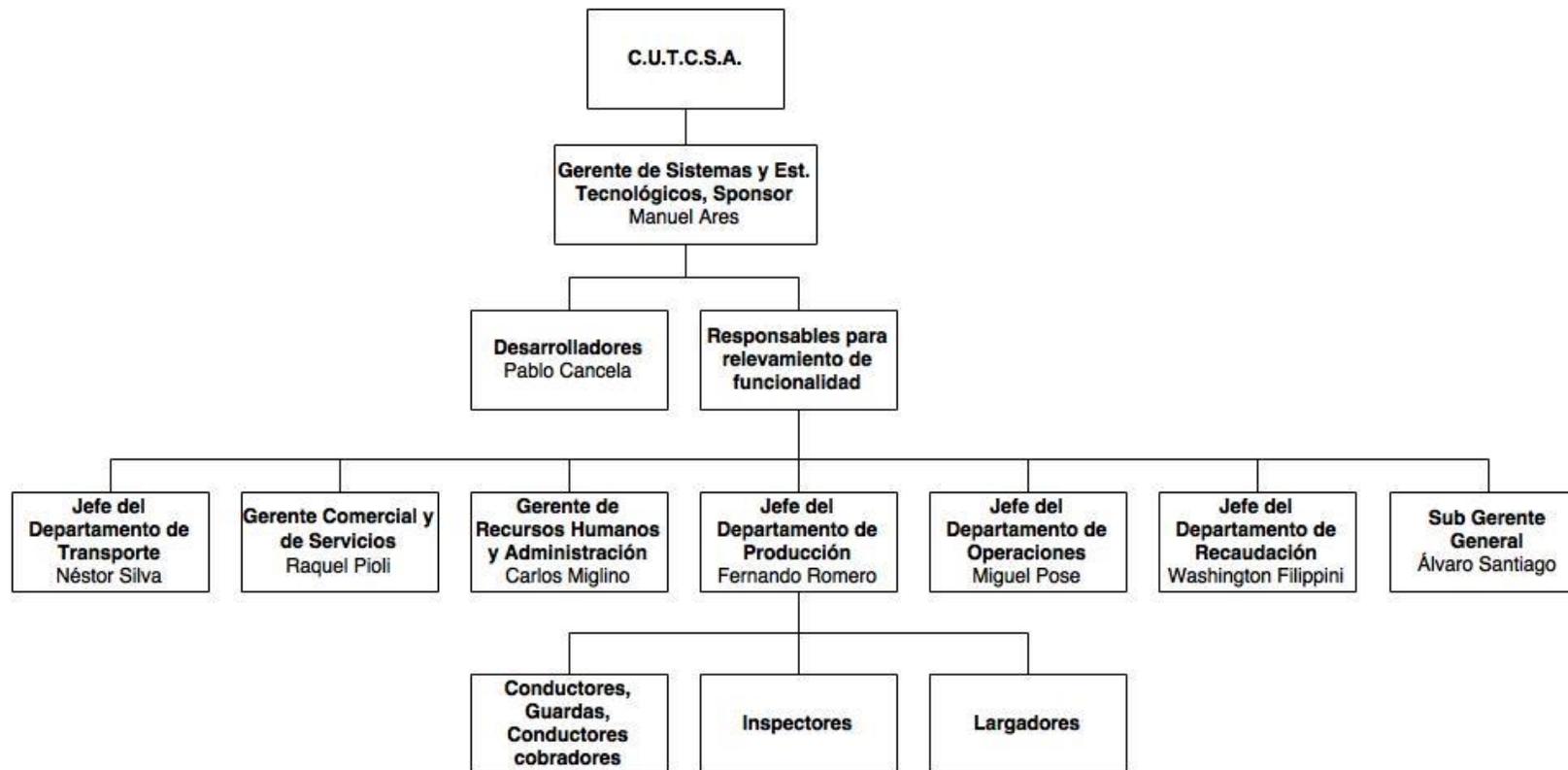


Ilustración 12-68 Stakeholders dentro de C.U.T.C.S.A

La conexión entre el proyecto y el *Sponsor* (C.U.T.C.S.A.) está dado por la siguiente relación:

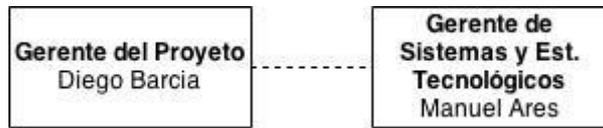


Ilustración 12-69 Relación entre el proyecto y la empresa

A continuación, se detalla la participación de cada interesado involucrado en el proyecto.

Equipo del proyecto		
Rol	Nombre	Responsabilidades
Tutor del Proyecto	Amalia Álvarez	Apoyo y guía para el desarrollo del proyecto.
Gerente del Proyecto	Diego Barcia	Gestión general del proyecto, la planificación, definición de la etapa inicial, elegir la metodología, hacer seguimiento del plan, y cierre del proyecto.
<i>Tester/</i> Desarrollador	Diego Barcia	Desarrollo Android, mantener el buen código, prácticas de estándares de programación, práctica de manejo de errores, prácticas para prueba unitarias, prácticas para refactorio.
Ingeniero de Requerimientos	Andrés Amegeiras	Definir la estrategia de relevamiento, encargado del relevamiento.
SCM	Andrés Amegeiras	Gestión de la configuración de todos los productos.
<i>Tester/</i> Desarrollador	Andrés Amegeiras	Desarrollo de la seguridad, mantener el buen código, prácticas de estándares de programación, práctica de manejo de errores, prácticas para prueba unitarias, prácticas para refactorio.
Arquitecto	Pedro Trecca	Solución técnica, documentar la arquitectura, documentar decisiones más relevantes y poder comunicarlo.
SQA	Pedro Trecca	Definir el proceso de trabajo, documentar cómo se va a trabajar, y definir y documentar el plan de aseguramiento de calidad.
<i>Tester/</i> Desarrollador	Pedro Trecca	Desarrollo del <i>backend</i> , Mantener el buen código, prácticas de estándares de programación, práctica de manejo de errores, prácticas para prueba unitarias, prácticas para refactorio.

Tabla 12-26 Responsabilidades de cada integrante del equipo del proyecto

Universidad ORT			
Rol	Nombre	Objetivos	Frecuencia
Tutor SQA	Amalia Álvarez	Apoyo para la gestión de la calidad	Ocasionalmente
Tutor Gestión Proyectos	Mariel Feder, Rafael Bentancur	Apoyo para la gestión del proyecto	Ocasionalmente
Tutor Test	Martín Solari	Apoyo para el desarrollo y gestión de pruebas	Ocasionalmente
Tutor Ing. Requerimientos	Álvaro Ortas, Marcelo Cagnani	Apoyo en desarrollo y gestión de requerimientos	Ocasionalmente
Tutor SQA	Leonardo Scafarelli	Apoyo para la gestión de la calidad	Ocasionalmente
Tutor Arquitectura	Gastón Mousques	Apoyo para el desarrollo de la arquitectura	Ocasionalmente

Tabla 12-279 Responsabilidades para el proyecto de los referentes dentro de la universidad

C.U.T.C.S.A.			
Rol	Nombre	Objetivos	Frecuencia
Gte. Sistemas y Est. Tecnológicos	Manuel Ares	Sponsor, aceptación producto	Varias veces por semana.
Desarrollador C.U.T.C.S.A.	Pablo Cancela	Aceptación desarrollo y arquitectura	Varias veces por semana durante el desarrollo de las evoluciones.
Jefe Dpto. de Transporte	Néstor Silva	Relevamiento, aceptación de su módulo del sistema	3 veces: al inicio del proyecto, para prototipación, y con el módulo desarrollado.
Jefe Dpto. de Producción	Fernando Romero	Relevamiento, aceptación de su módulo del sistema	Al inicio del proyecto.

Jefe Dpto. de Operaciones	Miguel Pose	Relevamiento, aceptación de su módulo del sistema	Al inicio del proyecto.
Jefe Dpto. de Recaudación	Washington Filippini	Relevamiento, aceptación de su módulo del sistema	Al inicio del proyecto.
Gte. Recursos Humanos	Carlos Miglino	Relevamiento, aceptación de su módulo del sistema	Al inicio del proyecto.
Gte. Comercial y Servicios	Raquel Pioli	Relevamiento, aceptación de su módulo del sistema	Al inicio del proyecto.
Sub Gte. General	Álvaro Santiago	Descripción del proyecto y producto de la empresa	Al inicio del proyecto, y al final de cada evolución.
Analista de Base de Datos	Daniel Barcia	Apoyo para la base de datos	Varias veces por semana durante la arquitectura.
Desarrollador	Guillermo López	Apoyo para el desarrollo	Varias veces por semana.
Analista en Redes	Federico Rena	Apoyo para el acceso a redes	Varias veces por semana

Tabla 12-28 Responsabilidades para el proyecto de los referentes dentro de la empresa

12.5.11. Dependencias Relevantes

A continuación, se describen las dependencias de distintos actores dentro del proyecto tenidos en cuenta para evitar retrasos y desvíos.

- Para poder comenzar con el diseño del primer ciclo, se necesitarán los requerimientos (completos y validados) del Gerente de Desarrollo y Estudios Tecnológicos, de lo contrario la falta de requerimientos puede generar una inconsistencia en el sistema y sus modificaciones traerán retrabajo que puede afectar seriamente el tiempo.
- Para poder comenzar con el diseño de la segunda evolución, se necesitarán los requerimientos (completos y validados) del Jefe de Producción, de lo contrario la falta de requerimientos puede generar una inconsistencia en el sistema y sus modificaciones traerán retrabajo que puede afectar seriamente el tiempo.
- El punto anterior se aplica para cada referente del módulo a desarrollar en cada ciclo.
- Existe dependencia por parte de Handsoft en caso de que usemos sus servicios y necesitemos realizar cambios en los mismos.

- También existe dependencia sobre la entrega del prototipo de las tarjetas que usarán los inspectores al equipo del proyecto, lo cual no permitirá terminar el desarrollo de dicha funcionalidad ni realizar pruebas.
- Disponibilidad del Gte. del Área de Sistemas y Est. tecnológicos, Analista en Redes, DBA y desarrolladores de la empresa para hacer el relevamiento de requerimientos no funcionales.
- La calidad, completitud y confiabilidad de los datos ofrecidos por la empresa, que consumirá nuestra aplicación, quedarán bajo responsabilidad de la empresa.

12.5.12. Gestión de comunicaciones

La tabla siguiente describe cada uno de los entregables a ser comunicados a los distintos *stakeholders* dentro del proyecto, y cómo los mismos son comunicados.

Entregable	Modalidad de Comunicación	Responsable a ser comunicado
Entrevistas	Correo electrónico, respondiendo con las correcciones	A quien se le realizó la entrevista
Descripción de la empresa y características principales del producto	Reunión	Sub Gte. General
Plan del proyecto	Reunión	Tutor de Gestión de Proyectos
Actualizaciones al plan del proyecto	Reunión	Gte. de Área de Sistemas
Plan de SCM	Reunión	Tutor de SCM
Plan de SQA y Calidad	Reunión	Tutor de SQA
Prototipo inspectores	Reunión	Jefe de Dto. de Producción e inspectores
ESRE general + inspectores	Reunión	Gte. de Área de Sistemas
ESRE general + inspectores	Correo electrónico, respondiendo con las correcciones	Sub Gte. General, Jefe Dto. de Producción, Desarrollador C.U.T.C.S.A.
Documento de Arquitectura	Reunión	Desarrollador C.U.T.C.S.A., Gte. de Área de Sistemas
Aplicación móvil general + inspectores	Reunión	Gte. de Área de Sistemas, Sub Gte. General, Jefe Dto. de Producción, Inspectores

ESRE conductores, guardas, conductores cobradores, largadores, común	Reunión	Gte. de Área de Sistemas
ESRE conductores, guardas, conductores cobradores, largadores, común	Correo electrónico, respondiendo con las correcciones	Sub Gte. General, Jefe Dto. de Producción, Desarrollador C.U.T.C.S.A.
Documento de Arquitectura	Reunión	Desarrollador C.U.T.C.S.A.
Documento de Arquitectura	Reunión	Gte. de Área de Sistemas
Aplicación versión final	Reunión	Gte. de Área de Sistemas, Sub Gte. General, Jefe Dto. de Producción
Documentos requeridos por la empresa	Reunión/correo electrónico	Gte. de Área de Sistemas
Documentos requeridos por la facultad	Reunión	Tutor del proyecto

Tabla 12-29 Tabla de productos comunicados

Se puede apreciar la estructuración de los principales productos del proyecto en el siguiente diagrama, separados por etapas.

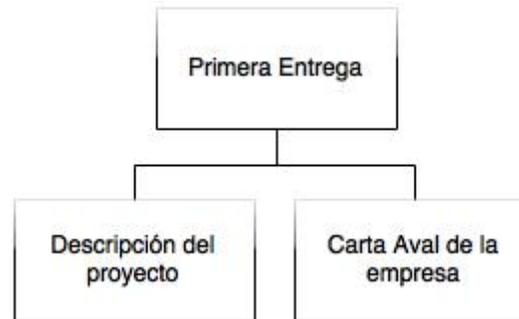


Ilustración 12-70 Primera entrega a realizar en la facultad

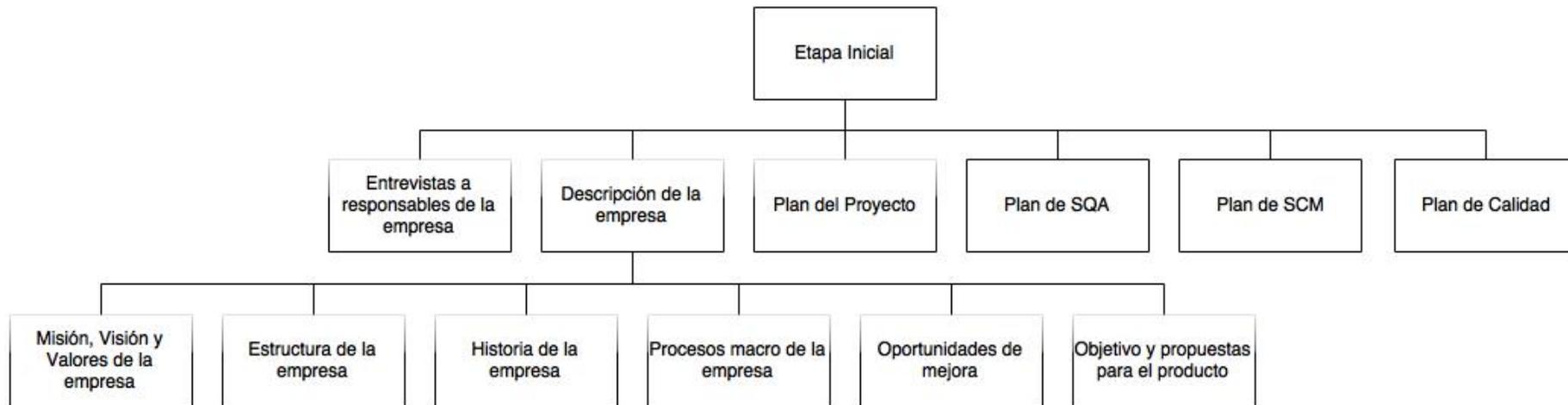


Ilustración 12-71 Productos correspondientes a la etapa inicial del proyecto

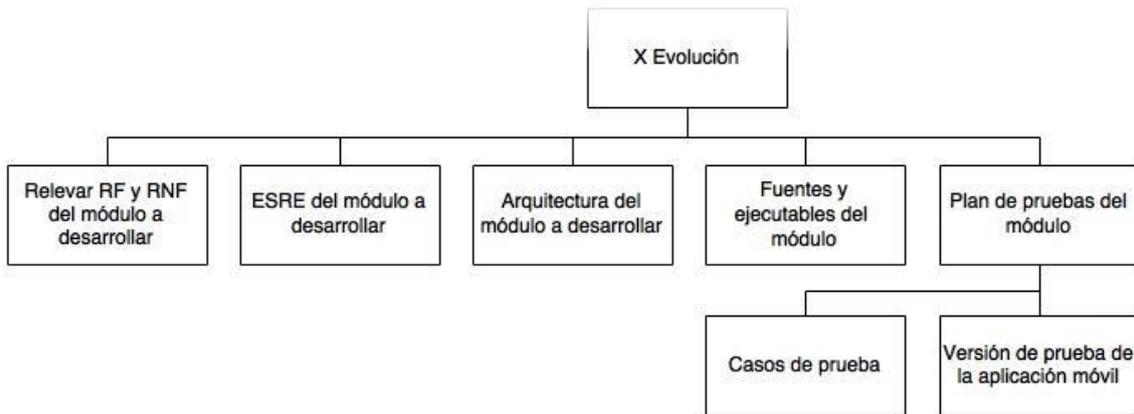


Ilustración 12-72 Productos correspondientes a cada evolución desarrollada

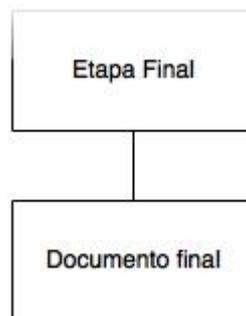


Ilustración 12-73 Productos correspondientes a la etapa final del proyecto

12.5.13. Supuestos, restricciones y obligaciones

A continuación, se detallan los supuestos, restricciones y obligaciones por parte del equipo para realizar las estimaciones correspondientes.

- La curva de aprendizaje del equipo podría ser considerada rápida, pero existen limitantes de tiempo a raíz de evaluaciones obligatorias, parciales y exámenes que los integrantes deben realizar en la facultad.
- Entre las obligaciones del equipo (así derechos que incluyen Licencias de trabajo) se encuentran:

Nombre	Descripción	Fecha inicio	Fecha fin	Probabilidad de que suceda	Disponibilidad para el proyecto
--------	-------------	--------------	-----------	----------------------------	---------------------------------

Diego Barcia	Entrega de obligatorio de Estrategia de Negocios	26/06/15	2/07/15	Seguro	Media
Pedro Treca	Examen de Redes	3/07/15	15/07/15	Seguro	Baja
Diego Barcia	Entrega de obligatorio GEIS	10/07/15	16/07/15	Seguro	Media
Pedro Treca	Examen de Gestión de Proyectos	16/07/15	28/07/15	Baja	Baja
Diego Barcia	Examen de Estrategia de negocios	24/07/15	30/07/15	Media	Media
Andrés Amegeiras	Exámenes	julio / agosto		Seguro	Baja
Diego Barcia	Presentación de Prueba de Software	22/07/15	28/09/15	Seguro	Alta
Pedro Treca	Examen de Gestión de Proyectos	8/09/15	23/09/15	Probable	Baja
Diego Barcia	Entrega proyecto (parte 1) Diseño de Interfaces	25/09/15	01/10/15	Seguro	Alta
Diego Barcia	Entrega proyecto (parte 2) Diseño de Interfaces	16/10/15	22/10/15	Seguro	Alta
Diego Barcia	Presentación de Diseño de Interfaces	29/10/15	04/11/15	Seguro	Alta
Diego Barcia	Presentación proyecto Diseño de Interfaces	05/11/15	11/11/15	Seguro	Alta
Diego Barcia	Entrega proyecto (parte 3) Diseño de Interfaces	06/11/15	12/11/15	Seguro	Alta

Diego Barcia	Entrega trabajos prácticos Prueba de Software	13/11/15	19/11/15	Seguro	Alta
Diego Barcia	Entrega trabajos prácticos Diseño de Interfaces	20/11/15	26/11/15	Seguro	Alta
Diego Barcia	Licencia	2/12/15	13/12/15	Seguro	Media
Diego Barcia	Parcial de Prueba de Software	02/12/15	09/12/15	Seguro	Media
Diego Barcia	Parcial de Diseño de Interfaces	09/12/15	14/12/15	Seguro	Baja
Andrés Amegeiras	Entregas y parciales	noviembre / diciembre		Seguro	Baja
Pedro Treca	Licencia	23/12/15	4/01/16	Seguro	Baja
Amalia Álvarez	Licencia	22/10/15	31/10/15	Seguro	Nula
Andrés Amegeiras	Licencia	05/01/16	18/01/16	Media	Media

Tabla 12-30 Obligaciones del equipo del proyecto a tener en cuenta

Disponibilidad	
Alta	no afectará/afectará muy poco en el proyecto
Media	podrá avanzar bastante en el proyecto
Baja	podrá avanzar poco en el proyecto
Nula	no podrá avanzar en el proyecto

Tabla 12-31 Valores sobre disponibilidad

Probabilidad	
Seguro	Seguro sucede
Alta	Muy Probable
Media	Probable
Baja	Poco probable

Tabla 12-32 Tabla de valores sobre probabilidad de que efectivamente ocurra

- Como restricción se encontraba la fecha límite de entrega del proyecto en la facultad, indicada para el 3/03/2016. También las entregas y revisiones intermedias planificadas para los meses de junio, agosto y noviembre.
- Se recomienda la dedicación de entre 15 y 20 horas semanales al proyecto por cada integrante. El equipo se propuso como obligación respetar dicha suma de horas, pero se tendrá en cuenta las obligaciones que existen en la facultad. Se usa como referencia unas 2,5 horas diarias por integrante para el cálculo de estimaciones.
- Se tomó como supuesto que se dispondrá de dispositivos móviles necesarios para realizar las pruebas y desarrollo.
- Se tendrá acceso a la empresa, durante horario laboral, para realizar desarrollos y pruebas del sistema.
- Las siguientes fechas han de ser tenidas en cuenta a la hora de la planificación y desarrollo del proyecto:

Feriados y Vacaciones		
Fecha Inicio	Fecha Fin	Descripción
19/06/15	21/06/15	Fin de semana largo a raíz del feriado del 19/6
29/06/15	12/07/15	Vacaciones de Invierno
22/08/15	25/08/15	Fin de semana largo a raíz del feriado del 25/8
19/09/15	27/09/15	Vacaciones de Primavera
Diciembre, enero y febrero		Vacaciones de Verano

Tabla 12-33 Feriados y vacaciones a tener en cuenta para la disponibilidad de stakeholders

- Todas las personas involucradas en el proyecto van a estar disponibles cuando se los necesite durante el horario laboral. Las mismas, a su vez, tienen los siguientes días de licencia, lo cual dificultará la reunión con ellos en esos períodos:

Nombre	Fecha Inicio	Fecha Fin	Descripción
Washington Filippini	06/07/15	12/07/15	Licencia
Washington Filippini	Setiembre (Posiblemente)		Licencia
Washington Filippini	Diciembre (Posiblemente)		Licencia
Fernando Romero	06/07/15	17/07/15	Poca disponibilidad
Pablo Cancela	18/11/15	05/12/15	Licencia
Federico Rena	Marzo		Licencia
Federico Rena	Noviembre		Indisponible
Daniel Barcia	Diciembre-Enero		Licencia

Tabla 12-34 Fechas en que los stakeholders no se encontrarán disponibles

- Horarios de tutores en la facultad, los cuales podremos solicitar su ayuda durante el desarrollo del proyecto:

Rol	Nombre	Horarios
SQA	Amalia Álvarez	Lunes 17:00 hs
Gestión de Proyecto	Mariel Feder	Martes o jueves 19:00 hs
Test	Martín Solari	Lunes 17:00 hs

Ing. Requerimientos	Alvaro Ortas	Martes y miércoles de 18:30 a 21:30 hs
SQA	Leonardo Scafarelli	Lunes de 16:00 a 20:00 18:30 a 21:30 hs
Ing. Requerimientos	Marcelo Cagnani	Miércoles y jueves de 18:30 a 21:30 hs
Gestión de Proyecto	Rafael Bentancur	Martes, miércoles y jueves de 19:00 a 21:00 hs
Arquitectura	Gastón Mousques	Lunes a jueves de 14:00 a 18:30 hs. Viernes de 9:00 a 13:00 hs.

Tabla 12-35 Horarios de disponibilidad de tutores en la facultad

12.5.14. Anexos

12.5.14.1. Cronograma del proyecto

Cronograma del proyecto, a través del análisis de puntos de función que se realizó para las distintas evoluciones del proyecto.



Ilustración 74 Diagrama de Gantt del proyecto

12.5.14.2. Análisis de otros riesgos

A continuación, se detallan los riesgos detectados al inicio del proyecto, pero que no se realizó seguimiento profundo.

Id	Descripción Riesgo	Plan de Respuesta	Plan de Contingencia	Estrategia del riesgo
1	Respuesta por parte de entrevistados para la	Especificar dos semanas para que manden una respuesta.	En caso de superar dicha fecha sin obtener respuesta, la decisión será	Eliminar

	corrección de entrevistas		realizada por el equipo del proyecto.	
2	Corrección y propuestas de prioridad para posibles módulos a desarrollar	Especificar dos semanas para que manden una respuesta.	En caso de superar dicha fecha sin obtener respuesta, la decisión será realizada por el equipo del proyecto.	Eliminar
3	Especificación correcta de la seguridad	1) Basarnos en sistemas de seguridad empleadas por la empresa. 2) Ayuda de expertos (en la empresa, en la facultad, etc.). 3) Buscar información.	Invertir más tiempo en análisis e investigación.	Mitigar
4	Diseño correcto de la seguridad	1) Basarnos en sistemas de seguridad empleadas por la empresa. 2) Ayuda de expertos (en la empresa, en la facultad, etc.). 3) Buscar información.	Invertir más tiempo en análisis e investigación.	Mitigar
5	Dificultad para reunirse con jefe de inspectores para mostrar prototipo	Avisar con anticipación las reuniones previstas	1) Dedicar el tiempo a otras tareas o adelantar trabajo hasta poder realizar la entrevista. 2) En caso de que el proyecto se tranque, hablar con el <i>Sponsor</i> para solución.	Mitigar
6	Falta de tiempo para completar documentación	Aumentar el esfuerzo	-	Mitigar

Tabla 12-36 Análisis de riesgos

A partir del impacto y la probabilidad de ocurrencia, se calculó la magnitud.

$$\text{Impacto} * \text{Probabilidad} = \text{Magnitud}$$

Id	Probabilidad de ocurrencia	Impacto	Magnitud
1	0.4	3	1.2
2	0.4	3	1.2
3	0.4	4	1.6
4	0.4	4	1.6
5	0.4	4	1.6
6	0,2	5	1

Tabla 12-37 Peso de cada riesgo dentro del proyecto

Impacto	
Valor	Descripción
0	Ninguno
1	Marginal
2	Poco importante
3	Importante (puede retrasar un poco el proyecto)
4	Muy Importante (puede retrasar mucho el proyecto)
5	Crítica (puede detener el proyecto)
6	Catastrófica (fracaso del proyecto)

Tabla 12-38 Valores del impacto tenidos en cuenta para el cálculo de la magnitud

Probabilidad de ocurrencia	
Valor	Descripción
0	Improbable
0.2	Poco probable
0.4	Probable
0.6	Muy probable
0.8	Altamente probable

1	Se convierte en problema
---	--------------------------

Tabla 12-39 Valores de la probabilidad de ocurrencia tenido en cuenta para el cálculo de la magnitud

12.5.14.3. Análisis de posibles requerimientos a desarrollar

Algunos requerimientos funcionales y propuestas posibles que se detectó al inicio del proyecto a través de entrevistas a distintos responsables dentro de la empresa se describen a continuación. La prioridad fue asignada a partir de una reunión con el *Sponsor*.

Área o departamento	Requerimiento funcional	Prioridad
Para todas las áreas de la empresa	Sincronizar el servicio de correo de la empresa en la aplicación móvil y así incentivar y facilitar la comunicación a través de este medio. Incluiría una mensajería interna.	Media
	Ver información sobre personal (cargo y función, tareas que realiza, horarios de trabajo, mail, ubicación del trabajo, etc.).	Media
	Mostrar ayuda sobre qué hacer paso por paso (ej. qué hacer a la hora de un choque, cómo realizar las tarjetas STM o pases libres, etc.). Poder ver distintos protocolos de trabajo, qué es lo que tiene que hacer el personal, cómo, y qué es lo que NO tiene que hacer.	Alta
	Mostrar novedades generales de la empresa: aumentos de sueldo, suba de boletos, feriados, jornadas de vacunación, cambios de horario, información actualizada y de fácil acceso sobre los beneficios que se pueden usar con la tarjeta "Bus", etc.	Alta
	Avisos y recordatorios: aviso que el sueldo ya fue acreditado en la cuenta bancaria del funcionario, aviso de excedentes o sobrantes de recaudación para los conductores y guardas, aviso que se encuentran disponibles los ticket de alimentación para ser retirados, vencimiento de carne de salud, presentarse en cierto lugar - citaciones, cambios de aceite, gomas, realizar citaciones para hacer declaraciones en RRHH, avisar cuándo es que tienen aprobada una licencia y que se les	Alta

	realice un recordatorio de cuándo tienen que reintegrarse a trabajar, falta de stock, etc.	
	Pedir reposición de insumos o reparaciones: rollos de boletos, papeles de impresión, reparación de computadora, reparación de fotocopidora, etc.	Baja
	Consultas y estadísticas de valores diarios, mensuales o anuales: consultas usadas en su tarea diaria, consultas de su recibo de sueldo o cuenta corriente, etc.	Alta
	Digitalizar el proceso correspondiente a tareas que tiene que ver con mandar un montón de planillas a otros sectores de la empresa, y así también evitar la duplicación de información.	Muy baja
	Envío y consulta de comunicados entre empleados o sectores de la empresa.	Baja
Departamento de Producción	Ver las paradas más cercanas a la ubicación del usuario y mostrar información sobre ella: número, dirección y líneas que pasan por ella.	Alta
	Mostrar en un mapa el recorrido que tiene cierta línea.	Alta
	Mostrar los próximos arribos de líneas a una parada, pudiendo filtrar por parada.	Alta
	Ver la ubicación e información (número, recorrido, destino) de cierto coche en un mapa.	Alta
	Conocer la ruta y qué ómnibus tomar desde un origen a cierto destino.	Alta
	Ver información de las oficinas que al personal le interesa saber (nombre, dirección, teléfono y horarios), y un acceso directo que muestre la ruta desde la posición del usuario hasta la misma.	Alta
	Obtener el reporte que en la actualidad se obtiene impreso a través de la expendedora de boletos, conteniendo la información necesaria para la inspección del servicio.	Alta
	Ver el horario teórico de cada recorrido para cierta línea.	Alta
	Poder controlar las unidades en tiempo real mediante una aplicación móvil.	Alta
	Mostrar los horarios de trabajo (cartones) que deben realizar conductores y guardas.	Alta
Cuando se realiza un cambio o interrupción en un servicio que no va a salir o va a ser reforzado, o rotura de un coche, que el mismo sea actualizado para que los horarios “teóricos” reflejen estos pequeños cambios	Media	

	que se sabe se van a dar (o al menos pueda ser advertido al funcionario o usuario).	
	Mostrar información sobre Servicios de coches: horarios asignados para un coche en cierto día/semana/mes; quién y horario que trabaja, qué coches están trabajando, etc.	Alta
	Función para pedido de auxilio de remolques para el funcionario.	Alta
	Función de botón de pánico.	Alta
Área Comercial y de Servicios	Ver el status de un coche dentro del taller (En espera, Ingresado, Pronto para retirar).	Media
	Ver agenda del centro de lubricación y alineación, y que pueda ser "autoagendable".	Media
	Mostrar información a propietarios y funcionarios del taller sobre reparaciones que se le hizo a un coche, cantidad de combustible que se le cargó, etc.	Media
	Mostrar información general del coche (servicios que realiza, quién es el propietario, matrícula, etc.)	Media
	Ver el análisis y estados del stock de componentes	Muy baja
	Incorporar la aplicación móvil ya existente, que permite encontrar al coche estacionado en el predio de la empresa, a la nueva aplicación.	Baja
	Poder analizar los códigos de fallas que salen en el tablero de los coches y describir soluciones o referencias al manual.	Media

Tabla 12-40 Análisis de posibles requerimientos a realizar

A su vez, se hizo un análisis de la prioridad para el desarrollo de los distintos módulos, descritos a continuación:

Área o Departamento	Módulo	Prioridad	Justificación
Todas	Funcionarios	Alta	Luego de tener a la mayoría de los funcionarios utilizando la app (gracias al módulo para guardas y conductores), esta funcionalidad será de mucha utilidad para todos ellos, y otros funcionarios que también

			quieran aprovechar las ventajas de este módulo.
Departamento de Producción	Propietarios	Media	Otra gran mayoría de la empresa son los propietarios de los coches.
	Inspectores	Alta	Pedido por la empresa
	Guardas y Choferes	Alta	La gran mayoría de los funcionarios de C.U.T.C.S.A. son guardas y choferes.
Área Comercial y de Servicios	Taller	Media	Apoyaría a funcionalidades de los módulos Funcionarios y Propietarios.
	Administración	Baja	Entre los últimos de la lista por indicación del cliente.

Tabla 12-41 Priorización de módulos para desarrollar

Para la fecha de finalización del proyecto de fin de carrera no se logrará cubrir toda la funcionalidad. Es por dicha razón que se desarrollarán sólo algunos módulos del sistema (comenzando por orden de prioridad y por la cantidad de potenciales usuarios), y el resto quedará como extensión del producto para fechas posteriores a la finalización del proyecto.

12.6. Anexo 6 - Plan de Calidad

ORTsf	Plan de la Calidad
<p style="text-align: center;">Aplicación de Apoyo Móvil C.U.T.C.S.A</p> <p>Objetivo</p> <p>El objetivo de este documento es definir las fases y actividades a realizar durante el proyecto para crear los entregables comprometidos con el cliente.</p> <p>Alcance</p> <p>Este documento es aplicable al proyecto Aplicación de Apoyo Móvil C.U.T.S.A y a las fases Identificación del sistema, Análisis, Arquitectura, Desarrollo, Prueba, Despliegue y a los procesos gestión, SCM, SQA.</p> <p style="text-align: center;">2015</p>	
Elaborado: Juan Pedro Trecca	10/7/2015

12.6.1. Metodología de trabajo

Se utilizará un ciclo de vida Evolutivo por lo que existirán varios ciclos de los cuales se obtendrán en cada fase un nuevo entregable de la fase correspondiente

12.6.2. Plan de la calidad

Nombre fase	Actividad	Producto Resultado	Producto/s Consumido/s	Rol Responsable	Roles Participantes	Documentos de referencia
Identificación del sistema	Estudio de las herramientas tecnológicas	Documento de investigación tecnológica	Información de diferentes tecnologías, necesidades del cliente	Arquitecto	Desarrolladores	Estándar de documentos internos
	Definición del negocio	Borrador del plan	Entrevistas a los responsables de Área	Analista de Requerimientos, Gerente	Cliente, Analista de Requerimientos, Gerente	ESRE mínimo
Análisis	Especificación de requerimientos	ESRE	Requerimientos, actores, necesidades del cliente	Analista de Requerimientos	Cliente, Analista de Requerimientos, Usuario	Plantilla de ESRE
	Validación de ESRE	ESRE validado	ESRE	Analista de Requerimientos	Cliente, Analista de Requerimientos	
Arquitectura	Diseño de la arquitectura de la aplicación servidor	Documento de arquitectura	ESRE	Arquitecto	Cliente, Arquitecto	Plantilla de Arquitectura
	Diseño de la arquitectura	Documento de	ESRE	Arquitecto	Cliente, Arquitecto	Plantilla de

	ra de la aplicación Móvil	arquitectura				Arquitectura
	Revisión de la arquitectura	Documento de arquitectura revisado	Documento de arquitectura	Líder de SQA	Arquitecto, juico experto	Estándar de documentos internos
Desarrollo	Codificación de la funcionalidad de una iteración de aplicación móvil	Fuentes y Ejecutables	ESRE, planificación de iteración	Líder de desarrollo	Desarrolladores	Estándar de código
	Codificación de la funcionalidad de una iteración de aplicación servidor	Fuentes y Ejecutables	ESRE, planificación de iteración	Líder de desarrollo	Desarrolladores	Estándar de código
Prueba	Diseño de casos de prueba	Planilla de casos de prueba	ESRE	Líder de Testing	Testers	Plantilla de diseño de casos de prueba
	Ejecución de pruebas	Resultado de ejecución de pruebas	Planilla de casos de prueba	Líder de Testing	Testers, desarrolladores, Arquitecto	Estándar de documentos internos
Despliegue	Despliegue de la aplicación	Informe de despliegue	Ejecutables	Arquitecto, Líder de Testing	Desarrolladores, Testers	Documento de despliegue
Procesos de Gestión de proyecto	Planificación de actividades	Cronograma de actividades	Plan de la calidad	Gerente de proyecto	Gerente, Líder de SQA	Estándar de documentos internos

	Gestión de riesgos	Plan de riesgos	ESRE, supuestos, Documento de investigación tecnológica	Gerente de proyecto	Desarrolladores	Estándar de documentos internos
	Diseño del Plan de la calidad	Plan de la calidad	Necesidades del proyecto, lista de entregables	Líder de SQA	Desarrolladores	Estándar de documentos internos
	Gestión de SCM	Plan de SCM	ESRE, supuestos, Documento de investigación tecnológica	Líder de SCM	Gerente, Líder de Arquitectura	Estándar de documentos internos
	Gestión de Comunicaciones	Plan de Comunicación	ESRE, supuestos, Documento de investigación tecnológica	Gerente de proyecto	Gerente, Líder de SQA	Estándar de documentos internos
	Gestión del Proyecto	Plan de Proyecto	ESRE, supuestos, Documento de investigación tecnológica	Gerente de proyecto	Gerente, Líder de Arquitectura	Estándar de documentos internos
	Gestión de SQA	Plan de SQA	Plan de la calidad, ESRE, supuestos, Documento de investigación	Líder de SQA	Gerente, Líder de SQA	Estándar de documentos internos

			ión tecnológica			
	Diseño del Plan de la calidad	Plan de la calidad	Necesidades del proyecto, lista de entregables	Líder de SQA	Desarrolladores	Estándar de documentos internos
	Control y seguimiento de actividades	Documento de avance de actividades	Cronograma de actividades	Gerente de proyecto	Gerente	Estándar de documentos internos

Tabla 12-42 Actividades de calidad

12.7. Anexo 7 - Plan de SQA

ORTsf	Plan de SQA
Aplicación de Apoyo Móvil C.U.T.C.S.A	
Objetivo El objetivo de este documento es generar el Plan de Aseguramiento de la Calidad del Software, SQA. Definir un conjunto de pautas que guíen las actividades de aseguramiento de la calidad a cargo del Líder de SQA. El Plan de SQA identifica los productos a evaluar, los criterios a utilizar, los estándares a aplicar, las actividades a realizar y las técnicas a utilizar en cada caso.	
Alcance Este documento es aplicable al proyecto Aplicación móvil de apoyo C.U.T.C.S.A y a las fases Identificación del sistema, Análisis, Arquitectura, Desarrollo, Prueba, Despliegue y a los procesos gestión, SCM, SQA.	
2015	
Elaborado: Juan Pedro Trecca	15/07/15
Revisado: Amalia Álvarez	28/12/15
Aprobado: Juan Pedro Trecca	13/02/16

12.7.1. Propósito

El objetivo de este documento es generar el Plan de Aseguramiento de la Calidad del Software, SQA. Definir un conjunto de pautas que guíen las actividades de aseguramiento de la calidad a cargo del Líder de SQA. El Plan de SQA identifica los productos a evaluar, los criterios a utilizar, los estándares a aplicar, las actividades a realizar y las técnicas a utilizar en cada caso.

12.7.2. Referencias

ANSI/IEEE. 1989. Std 730.1-1989, IEEE Standard for Software Quality Assurance Plans.

ANSI/IEEE. 2008. Std 1028-2008, IEEE Standard for Software Reviews and Audits.

ANSI/IEEE. 1993. Std 1059-1993, IEEE Guide for Software Verification and Validation Plans.

UNIT. 2005. UNIT-ISO 9000:2005, Sistemas de gestión de la calidad - Fundamentos y vocabulario.

12.7.3. Definiciones

- Revisión: actividad emprendida para asegurar la conveniencia, adecuación y eficacia del tema objeto de la revisión, para alcanzar unos objetivos establecidos. – UNIT- ISO 9000:2005.
- Validación: confirmación mediante aprobación de evidencia objetiva de que se han cumplido los requisitos para una utilización o aplicación específica prevista. – UNIT- ISO 9000:2005.
- Verificación: confirmación mediante aprobación de evidencia objetiva de que se han cumplido los requisitos especificados. – UNIT- ISO 9000:2005.
- SQA: Acrónimo de Aseguramiento de la Calidad del Software (Software Quality Assurance)
- SCM: Acrónimo de Gestión de la Configuración del Software (Software Configuration Management)

12.7.4. Gestión

En la siguiente tabla se detallan los responsables por el aseguramiento y control de la calidad en este proyecto. La organización del proyecto se detalla en el Plan de Proyecto.

Rol	Nombre	Responsabilidades
Líder de SQA	Juan Pedro Trecca	Es responsable por el aseguramiento de la calidad de los productos generados en el proyecto. Es responsable por la planificación, seguimiento y evaluación de las actividades de este plan. También por la ejecución de las actividades definidas en este plan, por la identificación de hallazgos referentes a la calidad del producto o del proceso y asegurar su resolución.
Tester	Juan Pedro Trecca, Andrés Amegeiras, Diego Barcia	Es responsable por la planificación y ejecución de las actividades de prueba del producto. Es responsable por la identificación de hallazgos referentes a la calidad del producto y asegurar su resolución.

Tabla 12-43 Roles en calidad

12.7.4.1. Organización

A continuación, se detallan las relaciones entre el líder de SQA con respecto a los integrantes del equipo de proyecto.

Relación con Gerente del proyecto

La relación con este es muy importante ya que implica que deben coordinar sus esfuerzos para cumplir las metas. Una de las responsabilidades del SQA es reportar el avance del proyecto al gerente, en los distintos aspectos referidos a la calidad.

Relación con el SCM

Entre estos se tiene que definir y ejecutar el circuito de control de cambios y verificar la integridad de los respaldos del repositorio, también definirán los estándares de la documentación y la disponibilidad de los distintos archivos y documentos.

Relación con el Arquitecto

Los estándares de calidad deben estar alineados con las decisiones arquitectónicas. Ya que se deben cumplir estos para que no haya problemas en los entregables. En el este caso el arquitecto y el líder de SQA es la misma persona.

Relación con el Ingeniero de Requerimientos

La relación con este será principalmente al inicio de cada ciclo, en la cual validará la solución que se propuso en la especificación de requerimientos.

12.7.4.2. Actividades y entregables

El Líder de SQA debe hacer un análisis de la criticidad del producto y determinar las actividades de aseguramiento de la calidad del software que se realizará.

En la siguiente tabla se debe identificar para cada entregable del proyecto, las actividades de QA que se realizarán para asegurar su calidad.

Entregable Asociado	Actividad
ESRE, Código fuente, Planes	Elaboración de estándares
Diagramas de procesos y de comunicaciones	Validación con el cliente
Descripción General de la empresa	Validación con el cliente
Producto de software y Documentos a entregar al cliente y a ORT.	Elaboración del Plan de SQA
Especificación de Requerimientos (ESRE)	Validación del ESRE
Especificación de Requerimientos (ESRE)	Verificación del ESRE
Documentación de arquitectura	Validación de los documentos
Documentos de proceso de Gestión	Validación de los documentos
Documentación de SCM	Validación de los documentos
Plan de Prueba	Verificación del Plan
Documentación Final	Revisión

Tabla 12-44 Actividades y Entregables

12.7.5. Estándares, prácticas y convenciones

12.7.5.1. Propósito

Identificar toda la documentación relativa a la gestión del proyecto, la arquitectura, el desarrollo, para la validación y verificación necesaria para asegurar que se cumple con el alcance definido y establecer los procedimientos para comprobar la consistencia de los documentos.

12.7.5.2. Estándares

A continuación, se listan los estándares que serán utilizados en éste proyecto. Los mismos fueron obtenidos de Internet y de trabajos anteriores, con la finalidad de utilizarlos como referencia para determinar que documentos deben existir y el contenido que debe tener cada uno de ellos.

Documentación	Estándares	Autor
ESRE	Estándar de ESRE	Definido por el equipo
Especificación de la arquitectura	Estándar de la Arquitectura, Documentos 302 y 303 de ORT	Definido por el equipo
Manual de Usuario	Estándar definido por el equipo	Definido por el equipo
Plan de Proyecto	Estándar de Plan de Proyecto	Definido por el equipo
Plan de SQA	Estándar de Plan de SQA	ORTsf
Documentación Final	Documentos 302 y 303 de ORT	
Condigo	Buenas Practicas de codificación	Definido por el equipo
Plan de Pruebas	Estándar de Plan de Pruebas	Definido por el equipo

Tabla 12-45 Documento y estandares

12.7.5.3. Producto de software

Codificar siguiendo estándar del producto de software que siga la empresa

12.7.5.4. Documentos Escritos

El estándar definido para la presentación de los documentos escritos se encuentra en el *template* (*Template* de documentación.docx), el cual tiene determinado el tamaño y formatos de títulos y textos, formas de enumeración, de presentación de tablas, índices, caratulas y demás aspectos que hacen al formato y presentación de un documento.

12.7.6. Revisiones

A continuación, se detallan las revisiones que se deberán realizar de al producto. Esta lista contiene qué productos que se desean revisar, contra que criterio y técnica a usar.

Producto a evaluar	Criterios / Técnica	Responsable	Registro de resultados	Cuando se realiza
Plan de proyecto	Validación con el cliente	Gerente de proyecto	Plan actualizado	Al finalizar el plan para cada ciclo
ESRE	Validación con el cliente	Ingeniero de Requerimientos	Informe de revisión y actualización en el ESRE con los cambios identificados	Cuando el documento tenga los requerimientos funcionales y no funcionales bien definidos
Documento de Arquitectura	Compleitud con los Requerimientos no funcionales del ESRE	Ingeniero de Requerimientos	Informe de revisión	Cuando estén los diagramas de prontos y explicada la relación
Interfaz aplicación	Juicio de experto	Ingeniero de Requerimientos	Informe de revisión	Una vez que exista una primera instancia de
Plan de Pruebas	Compleitud con los requerimientos funcionales del ESRE	Líder SQA	Informe de revisión	Cuando las pruebas correspondientes estén finalizadas.

Tabla 12-46 Revisiones del producto

12.7.7. Validaciones

En este capítulo se especifica qué productos serán validados por el cliente, cuáles serán los criterios objetivos de la validación a utilizar, quién es responsable de realizar la validación, qué registro se dejará como evidencia de la validación y en qué momento será realizada la misma.

Producto a evaluar	Criterios / Técnica	Responsable	Registro de resultados	Cuando se realiza
Interfaz gráfica	Prototipación	Ingeniero de Requerimientos	Informe de validación	Luego de que el ESRE esté terminado.
Interfaz gráfica	Idem	Líder de SQA	Acta de reunión	Luego de que el producto esté terminado.
Plan de Proyecto	Idem	Líder de SQA	Acta de reunión	Idem
Informe de revisión	Idem	Líder de SQA	Acta de reunión	Idem
ESRE	Idem	Líder de SQA	Acta de reunión	Idem
Cierre del Proyecto	Idem	Líder de SQA	Acta de reunión	Idem

Tabla 12-47 Validaciones de producto

12.7.8. Pruebas

Se definió un **Plan de Pruebas** que contiene todas las actividades a realizar. A continuación, se muestra un resumen de las secciones más importantes del plan.

Estrategia de Test del Sistema

Se deberán realizar pruebas unitarias, pruebas funcionales y pruebas de integración.

Pruebas Unitarias:

Las pruebas unitarias las realizará cada encargado de desarrollo, solo se probarán las funcionalidades más críticas. Estas pruebas solo se realizarán para el backend de la aplicación.

Las pruebas unitarias se implementan utilizando el framework NUnit.

Pruebas Funcionales:

Para las pruebas funcionales utilizaremos la técnica basada en Casos de Uso, donde se verificará la correcta implementación de los flujos básicos y alternativos de todos los casos de uso a implementar en la iteración.

Pruebas de Integración:

Estas pruebas servirán para verificar la correcta integración entre el backend y la aplicación móvil.

Los casos de pruebas se crearán indicando los siguientes puntos:

- ID CP.
- Fecha de Prueba.
- Responsable.
- Objetivo.
- Precondición.
- Descripción del caso.
- Caso de Uso.
- Resultados Esperados.
- Resultados Obtenidos.
- Descripción del error.

Para estas se deberá crear una planilla en donde se creará una página de inicio en donde estarán los resultados de ejecución de las pruebas y en las siguientes paginas las pruebas realizadas. Cada prueba debe estar detallados sus pasos y los resultados de su ejecución.

Pruebas de sistema:

Son similares a las de integración, solo que estas buscan probar el sistema como un todo. Están basadas en los requerimientos generales y abarca todas las partes combinadas del sistema.

Para realizarlas se deberá acordar con el cliente como se podrán realizar, cuales son los requisitos que se necesitan para que se pueda ejecutar pruebas de sistema correctamente.

Criterios de resultados de prueba

Los criterios que pueden ser asignados a los resultados de prueba en un Caso de Prueba son los siguientes:

- **Exitosa:** El resultado de la prueba corresponde a lo esperado
- **Parcialmente exitosa:** El resultado de la prueba contiene errores menores que no impiden el uso del sistema y el mismo es parcialmente incorrecto.
- **Errónea:** El resultado de la prueba es completamente incorrecto o impide continuar el uso del sistema.

En caso de que la ejecución de un caso de prueba presente errores, tanto críticos como no críticos, este debe ser ingresado como un incidente en Jira como lo está indicado en el siguiente punto de este plan.

12.7.9. Reporte de problemas y acciones correctivas

Para reportar los errores se utilizará la herramienta de Atassilan, Jira, ya que encontramos que es una herramienta fácil de utilizar e intuitiva en la cual se pueden visualizar reportes.

Al registrar un problema en la herramienta se indicará la prioridad del mismo, el estado del problema. También contará con un nombre, que deberá ser descriptivo, y una descripción detallada del problema y en caso de ser necesario incluir imágenes para aclarar el problema.

Todo problema debe tener al menos un responsable, pero existe la posibilidad de que se asignen más de 1 responsable.

La herramienta permite realizar comentarios que serán utilizados para agregar sugerencias para resolución del problema y para comentar sobre avances en el problema.

Se tomarán solo 3 niveles de severidad de los ofrecidos por la herramienta estos son:

- **High:** cuando el problema hace que el sistema se caiga o impida el uso de las demás funcionalidades.
- **Medium:** cuando el requerimiento no funciona o no cumple con la funcionalidad básica esperada para el mismo

- Low: cuando el problema refiere a un comportamiento errático de la aplicación, pero no implica que el requerimiento cumpla con su funcionalidad. También cuando existen problemas con la interfaz gráfica se deben reportar estos errores.

Los problemas tendrán 3 estados:

- Por Hacer: Este estado corresponderá a los problemas que se crearan y todavía no se empezaron o problemas que se han vuelto a abrir en caso de que se vuelvan a encontrar errores.
- En Progreso: Este estado indica que se está trabajando en el problema
- Cerrado: Los problemas solo se podrán cerrar luego de que el mismo haya sido probado.

Como todos los participantes del proyecto serán al mismo tiempo desarrolladores y ejecutarán las pruebas no hay un responsable se idéntica un defecto. La persona que identifica un defecto, deberá subirlo a Jira y asignar un responsable. El responsable será la persona que desarrollo la funcionalidad (este puede ser el mismo que identifico el problema). Luego se asignará a otra persona para que pruebe el cambio, la persona asignada a realizar el cambio no puede ser la misma que la persona que probará el mismo.

A continuación, se incluye un diagrama de flujo que modela las actividades, secuencia y responsables para la detección, registro y resolución de problemas.

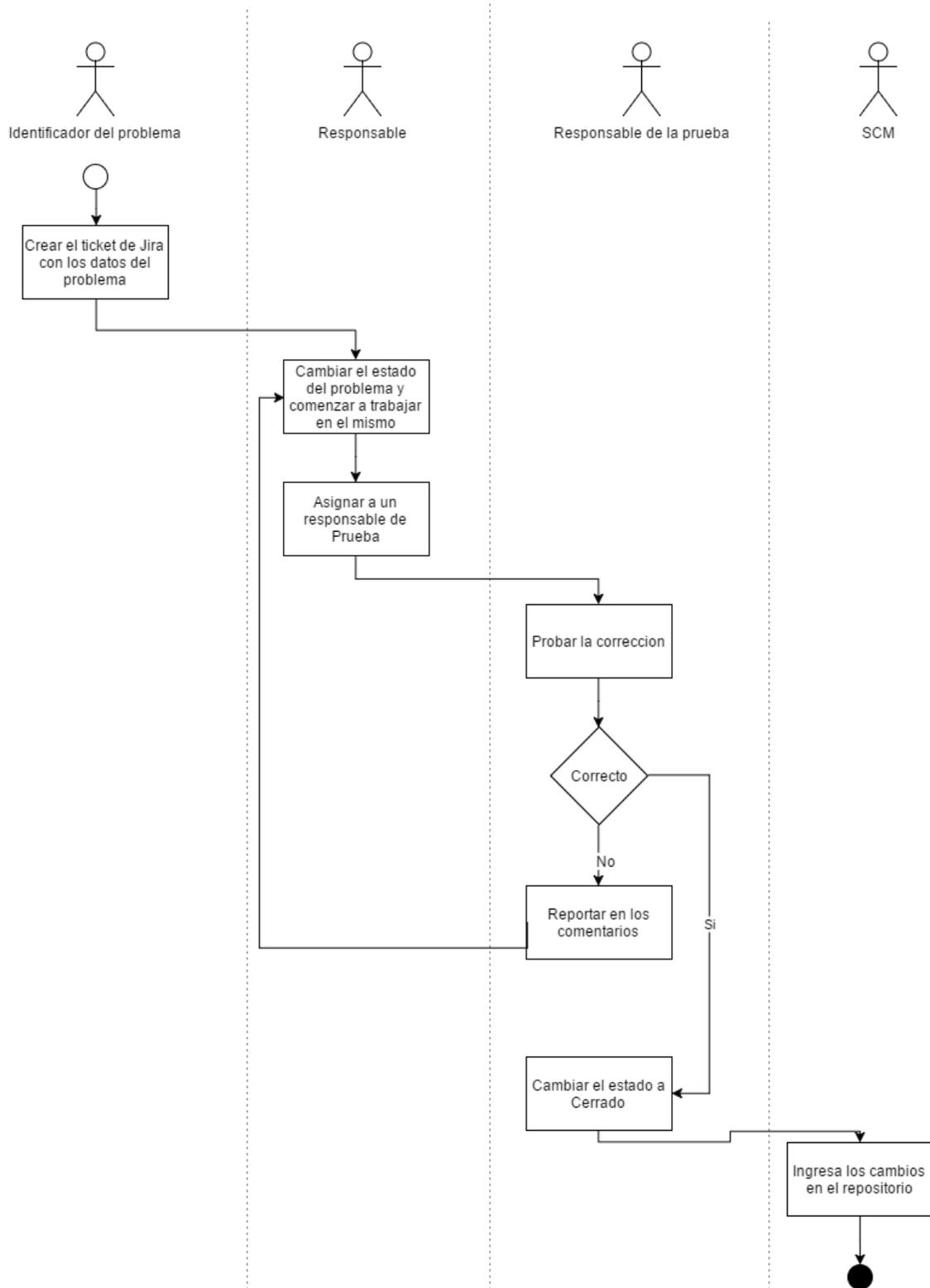


Ilustración 12-75 Reporte problema

12.7.10. Herramientas, técnicas y metodologías

Las herramientas a usar en el desarrollo del producto son:

- Entorno de desarrollo backend- .Net.
- Entorno de desarrollo móvil - Android SDK (Android Studio como IDE).
- Procesador de texto - Microsoft office Word y Google Docs.
- Planillas -Google Sheets.
- Reporte de Incidentes - Jira, Atassilan
- Gestión de Tareas - Manage It (Web, Google Play, Apple Store)
- Administración del proyecto - Microsoft Project.
- Repositorio de desarrollo – Git y Microsoft Foudation Server
- Repositorio general – Google Drive.

12.7.11. Control de código y de medios

La tarea de SQA en cuanto a control de código, es asegurar el cumplimiento de lo estipulado en el Plan de SCM

12.7.12. Formación

Para el entrenamiento en Android se realizó un plan con distintos tipos de ejercicios con fechas límites con el objetivo de familiarizarse con la herramienta Android Studio y con funcionalidades similares a las cuales se desarrollan.

12.7.13. Anexos

12.7.13.1. Buenas Prácticas de Programación

Este es un resumen de las practicas básicas que se deberán seguir cuando se esté programando. Se darán prácticas que apliquen para cualquiera de los lenguajes y luego se especificarán para lenguaje respectivamente.

El objetivo de esto es:

- Estandarizar el desarrollo
- Establecer reglas.
- Hacer el código fácil de leer
- Facilitar la reutilización
- Lograr escalabilidad en el código [43]

12.7.13.1.1. Practicas Generales

Evitar repetirse

Con esto se busca la reutilización del código ahorrando tiempo y dejando el código más fácil de entender.

Código Simple

Si bien esto puede ser difícil de conseguir, es deseable que el condigo sea lo más simple posible. Esto se puede lograr comentando el código que ayuda a la comprensión de los algoritmos.

Evitar Código Basura

En el caso de que un método, clase o variable, haya quedado obsoleto ya sea porque el mismo fue creado para probar algo, pero no la funcionalidad deseada o porque se encontró una mejor solución, es recomendable que se elimine para evitar confusiones y hacer de la solución general más difícil de comprender.

Nombres nemotécnicos

A la hora de declarar clases, variables, métodos, o cualquier elemento del código a utilizar, el nombre del mismo debe corresponder a la función que va a cumplir. De esta forma se logrará evitar confusiones y será más fácil trabajar con los mismos.

Responsabilidad Única

Los métodos deben responder a una única funcionalidad, que debe estar propiamente definida para ese método en el cabezal del mismo

Interfaces

Tanto Java con C# son lenguajes que soportan el uso de interfaces que serán usadas para ocultar los detalles de implementación.

Try-Catch

Se deben evitar escribir códigos try y catch dejando el bloque de catch vacío. Estos siempre deben de tener una acción.

12.7.13.1.2. Documentación y comentarios de código para C#

Comentarios de líneas de código particulares se realizarán una línea por encima de la misma y no tendrán definido un formato específico [44].

Por otro lado, previo a cada declaración de clase y método se comentará distinta información importante relevante para ayudar a desarrolladores a entender más fácilmente el código. Para ello se utilizarán los comentarios con triple retro barra “/” y en formato XML. Por lo que además de quedar los comentarios visibles en el código, esto permitirá ver los comentarios del método o clase escritos cuando estos sean invocados en cualquier otra parte del código. Además, se generará un archivo .XML por cada *namespace* del sistema y quedando guardado en este todos los comentarios y su método o clase asociados [42].

Los *tags* XML para realizar los comentarios en el código serán los siguientes:

summary: Se usará para describir de forma resumida que es lo que hace la clase o método.

param: Se usará para describir los distintos parámetros de cada método.

returns: Se usará para describir lo que el método retorna.

Ejemplo de comentario en un método:

```
/// <summary>
/// Este metodo hace la suma entre el número a y el número b
/// </summary>
/// <param name="a">primer parametro a sumar</param>
/// <param name="b">segundo parametro a sumar</param>
/// <returns>se devuelve el resultado de la suma</returns>
public int Suma(int a, int b)
```

Ilustración 12-76 Ejemplo de comentario

Luego de generado el archivo .XML en caso de necesitar navegarlo o simplemente para interpretarlo correctamente, se agrega automáticamente al comienzo del nombre de cada miembro una letra la cual identifica el tipo del mismo, siguiendo el siguiente estándar:

T: Indica que es de tipo: *class*, *interface*, *struct*, *enum* o *delegate*.

M: Indica que es un método.

Fragmento del archivo XML generado, siguiendo el ejemplo anterior del comentario en un método:

```
</member>
<member name="M:AccesoADatosGeneral.AccesoADatosGeneral.Suma(System.Int32,System.Int32)">
  <summary>
    Este metodo hace la suma entre el número a y el número b
  </summary>
  <param name="a">primer parametro a sumar</param>
  <param name="b">segundo parametro a sumar</param>
  <returns>se devuelve el resultado de la suma</returns>
</member>
```

Ilustración 12-77 Ejemplo de XML

12.7.13.1.3. Documentación y comentarios de código Java en Android

Envío de objetos entre actividades (*Activity*) a través de *Bundle*: a través del *put*, pasar un recurso ingresado en el archivo `bundle.xml` (Ver descripción de la documentación del `bundle.xml` en la Documentación y comentarios de código XML en Android).

Correcto:

```
bundle.putBoolean(activity.getString(R.string.bundle_boolean_que_indica_si_es_true_un_elemento_del_ui), true);
```

Incorrecto:

```
bundle.putBoolean("bundle_boolean_que_indica_si_es_true_un_elemento_del_ui", true);
```

Nombre de atributos de clase:

- empiezan con “m”.
- cada nueva palabra referente al atributo se comienza con su primera letra en mayúsculas, seguido con el resto de la palabra en minúsculas. No usar caracteres especiales (ñ, tildes, etc.).
- Excepción: Si es un atributo final, ver cómo se ingresan los nombres de atributos final.

Correcto: `mNombrelImagen`

Incorrecto: `nombrelImagen`, `mID`, `ID`, `mNombre_Imagen`

Nombre atributos final:

- todos los caracteres en mayúsculas.
- separación entre palabras con “_”.
- no usar caracteres especiales (ñ, tildes, etc.).

Correcto: `NOMBRE_IMAGEN`

Incorrecto: `NOMBREIMAGEN`, `NombrelImagen`, `Nombre_Imagen`

Nombre de atributos de métodos (tanto en parámetro como internos):

- empiezan con mayúsculas.
- cada nueva palabra referente al atributo se comienza con su primera letra en mayúsculas, seguido con el resto de la palabra en minúsculas. No usar caracteres especiales (ñ, tildes, etc.).
- Excepción: Si es un atributo final, ver cómo se ingresan los nombres de atributos final.

Correcto: idImagen

Incorrecto: mIdImagen, idimagen, IDimagen, id_imagen, ID_IMAGEN, IDIMAGEN

Nombre de clases:

- empiezan con mayúsculas.
- cada nueva palabra referente a la clase se comienza con su primera letra en mayúsculas, seguido con el resto de la palabra en minúsculas. No usar caracteres especiales (ñ, tildes, etc.).

Correcto: MiClase

Incorrecto: miClase, MiClase, Mi_Clase, MI_CLASE, CLASE

Nombre de métodos:

- empiezan con minúsculas.
- cada nueva palabra referente al método se comienza con su primera letra en mayúsculas, seguido con el resto de la palabra en minúsculas. No usar caracteres especiales (ñ, tildes, etc.).

Correcto: miMetodo

Incorrecto: MiMetodo, Mimetodo, mi_metodo, MI_METODO, METODO

Comentarios del código

Para los comentarios se usa la notación `/** */`. [45]

Los *tags* para realizar los comentarios en el código serán los siguientes:

@author: Se usará para indicar el autor la clase.

Autor: Se usará para indicar el autor de cada método.

Fecha: Se usará para indicar la fecha de creación de cada método y clase.

@param: Se usará para describir los distintos parámetros de cada método.

@return: Se usará para describir lo que el método retorna.

@see: Se usará para indicar una referencia para más información sobre una clase, método o atributo. Su uso es opcional.

Comentarios de Clases

Ítems obligatorios:

- Primera línea (o párrafo) para una descripción del rol de la clase.
- Segunda línea para la fecha de creación.
- Tercera línea para el autor de la clase, usando el Tag **@author**.

```
/**
 * Clase encargada de realizar alguna funcionalidad en el sistema.
 * Fecha: 28/08/2015
 * @author Diego Barcia
 */
public class MiClase{

}
```

Ilustración 12-78 Comentario en JAVA

Comentarios de Métodos

Ítems obligatorios:

- Primera línea (o párrafo) para una descripción del método.
- Segunda línea para el autor del método.
- Tercera línea para la fecha de creación.
- Una línea por cada parámetro del método (siempre y cuando tenga uno), usando el Tag **@param**.
- Una línea para el retorno del método (siempre y cuando tenga uno), usando el Tag **@return**.

```

/**
 * Suma el número sumando1 y el número sumando2.
 * Autor: Diego Barcia
 * Fecha: 28/08/2015
 * @param sumando1 primer parámetro a sumar.
 * @param sumando2 segundo parámetro a sumar.
 * @return el resultado de la suma.
 */
public int suma(int sumando1, int sumando2){
    return sumando1 + sumando2;
}

```

Ilustración 12-79 Comentario con Parámetros

12.7.13.1.4. Formato JSON para comunicación de servicios

Los datos enviados en la comunicación entre el *backend* y el *frontend* se realizará utilizando JSON, por lo que se definió una estructura común que los mismos deberán tener. Se detallan a continuación los distintos atributos que debe

Método: nombre de la operación

Estado: Correcto, Error_Interno: cae app, Error_datos: parámetros inválidos

Datos: la respuesta al método

12.7.13.1.5. Documentación y comentarios de código xml en Android

Comentarios

Para los comentarios se usa la notación `<!-- -->`

En el atributo *name* del xml:

- todos los caracteres en minúsculas.
- separación entre palabras con “_”.
- no usar caracteres especiales (ñ, tildes, etc.).

Para cada archivo .xml, se debe poner una descripción de qué tipos de valores se almacenan en dicho archivo, ingresando los siguientes valores como comentario:

Descripción: indica qué tipo de valores se almacenan en dicho archivo.

Fecha: fecha de creación del archivo.

Autor: el autor que creó el archivo.

Valores *float*

Para los casos de valores *float*, como no hay un *resource* XML para *float*, se utiliza la palabra "*float*" al inicio del *name*, y al usuario se sabrá que se puede castear a *float*.

Descripciones de mensajes

Para ingresar valores referentes a descripciones de mensajes a mostrar al usuario:

- utilizar *string*
- colocarlo en el archivo mensajes.xml
- el *tag name*
- todos los caracteres en minúsculas.
- separación entre palabras con "_".
- no usar caracteres especiales (ñ, tildes, etc.).
- si es una descripción de reporte, el tag name debe empezar con "reporte_" y continuar con una palabra referenciando el error.
- si es un título del *popup* de mensaje, el *tag name* debe empezar con "titulo_mensaje_" y continuar con una palabra referenciando el título del mensaje.

Tipos de reportes

Para ingresar valores referentes a los tipos de reportes a mostrar al usuario:

- utilizar *string*
- colocarlo en el archivo reportes.xml
- el *tag name*
- todos los caracteres en minúsculas.
- separación entre palabras con "_".
- no usar caracteres especiales (ñ, tildes, etc.).

- si es una descripción de reporte, el *tag name* debe empezar con “reporte_” y continuar con una palabra referenciando el reporte.
- si es un título del menú de reporte, el *tag name* debe empezar con “titulo_reporte_” y continuar con una palabra referenciando el título de reporte.

Descripciones de errores

Para ingresar valores referentes a descripciones de errores a mostrar al usuario:

- utilizar *string*
- colocarlo en el archivo errores.xml
- si es una descripción de error, el *tag name* debe empezar con “error_” y continuar con una palabra referenciando el error.
- si es un título del *popup* del error, el *tag name* debe empezar con “titulo_error_” y continuar con una palabra referenciando el título de error.

Valores para colores:

- utilizar **color**
- Colocar en el archivo colores.xml
- Como *name* usar
 - todos los caracteres en minúsculas.
 - separación entre palabras con “_”.
 - no usar caracteres especiales (ñ, tildes, etc.).
 - el nombre del color:
 - el color básico (primario, secundario, terciario, negro, blanco), usar su nombre. Ejemplo:

```
<color name="azul">#FF2196F3</color>
```

- tonalidades de color, usar como nombre la gama (desde más oscuro a más claro):
 - “oscuro_extremo”
 - “mas_oscuro”
 - “poco_mas_oscuro”
 - “oscuro”
 - “claro”
 - “poco_mas_claro”
 - “muy_claro”
 - “mas_claro”
 - “claro_extermo”

Ejemplo:

`<color name="azul_claro_extremo">#FFE3F2FD</color>`

- si tiene transparencia:
 - “transparente”

Ejemplo:

`<color name="azul_muy_claro_transparente">#CC90CAF9</color>`

- El nombre del elemento UI donde se aplica el color. Ejemplo:

`<color name="fondo_lista_claro_transparente">#AA42A5F5</color>`

Mensajes de carga

Para ingresar valores referentes a descripciones de cargas a mostrar al usuario:

- utilizar *string*
- colocarlo en el archivo `cargas.xml`
- el *tag name*
- todos los caracteres en minúsculas.
- separación entre palabras con “_”.
- no usar caracteres especiales (ñ, tildes, etc.).
- si es una descripción de carga, el tag *name*: debe empezar con “carga_” y continuar con una palabra referenciando qué carga.
- si es un título del *popup* de carga, el tag *name* debe empezar con “titulo_error_” y continuar con una palabra referenciando qué carga.

Claves *Bundle*

Para ingresar valores referentes a los *Bundle’s keys*:

- utilizar *string*
- colocarlo en el archivo `bundle.xml`
- el tag **name**
 - debe empezar con “bundle_” y continuar con una palabra referenciando el *Bundle’s keys*.
 - todos los caracteres en minúsculas.
 - separación entre palabras con “_”.
 - no usar caracteres especiales (ñ, tildes, etc.).
- El cuerpo del *string* no importa su valor.

Ejemplo:

`<string name="bundle_enviar_mensaje">enviarMensaje</string>`

Resources

Para recursos (*resource*) que se utilizan en un módulo o clase en particular, crear un nuevo archivo .xml:

- nombre del archivo:
 - todos los caracteres en minúsculas.
 - separación entre palabras con “_”.
 - no usar caracteres especiales (ñ, tildes, etc.).
- nombre de los recursos que se definen en el archivo:
 - comienza con el nombre del archivo, se agrega “_”, y se continúa con un nombre descriptivo del recurso.
 - todos los caracteres en minúsculas.
 - separación entre palabras con “_”.
 - no usar caracteres especiales (ñ, tildes, etc.).

12.8. Anexo 8 - Informe Primera Revisión

Grupo:	Proyecto para C.U.T.C.S.A. Andrés Amegeiras Diego Barcia Pedro Trecca		Fecha:	27/06/2015
Carrera:	Ingeniería en Sistemas		Revisores:	Amalia Álvarez Gastón Mousques

12.8.1. Fortalezas del Grupo

- Proceso de Investigación del negocio para conocer el contexto de la empresa y del problema y resultado obtenido: Procesos y Mapa de comunicaciones.
- Actitud de realizar prototipos.

12.8.2. Oportunidades de Mejora

- El nombre del proyecto.
- No queda completamente claro de qué trata el producto. Y explicar mejor a quién estamos resolviendo el problema.
- Analizar otros enfoques o posibilidades más allá de una aplicación móvil.
- Aclarar quiénes son los más interesados y quienes tienen mayor necesidad.
- Describir mejor qué son los módulos, de qué tratan.
- Definir prioridad de módulos a desarrollar.
- Aclarar mejor las restricciones o requerimientos no funcionales del producto.
- Identificar requerimientos no funcionales (no solo a TI, sino también a usuarios finales, los cuales nos podrán decir sobre qué datos específicos son sensibles y cuáles no para tenerlos en cuenta para la seguridad).
- Comunicar al cliente el Plan del proyecto para que estén en conocimiento sobre cuándo se va a desarrollar un módulo y que estén disponibles todas las dependencias relevantes para poder llevar a cabo las tareas (personas para relevamiento, otros sistemas que nos proporcionen los datos, etc.).
- Tener en cuenta herramientas alternativas para el registro de tareas a realizar.
- Estamos trabajando con mucha incertidumbre, con planes a muy alto nivel, sin bajar cosas a tierra.
- Evaluar la necesidad de incluir la primera evolución identificada referido al desarrollo de la arquitectura. En caso de no ser necesario, comenzar con el primer módulo identificado (módulo para Inspectores).

12.8.3. Acciones a Realizar

#	Descripción de la Acción	Fecha estimada
1	Reuniones para determinar requerimientos no funcionales y restricciones.	03/07/2015
2	Realizar reuniones para mostrar el prototipo para inspectores y comenzar el proceso de desarrollo.	Al inicio de la evolución para inspectores (nace de la evaluación de la necesidad de la arquitectura)
3	Identificar todos los interesados y los que tienen más necesidad.	10/06/2015
4	Analizar herramientas alternativas para registro de tareas	29/06/2015

Tabla 12-48 acciones a realizar primera revision

12.9. Anexo 9 - Informe Segunda Revisión

Grupo:	Proyecto para C.U.T.C.S.A. Andrés Amegeiras Diego Barcia Pedro Trecca		Fecha:	13/09/2015
Carrera:	Ingeniería en Sistemas		Revisores:	Amalia Álvarez Martín Solari

Fortalezas del Grupo

- Decisión de implementar como solución una aplicación móvil es acertada según usuarios objetivo y contexto donde va a ser utilizada.
- Grandes avances en Ingeniería de Requerimientos, es muy positivo la utilización de modelados.
- Presentación del proyecto correcta, se siguió una línea coherente y se mostraron todas las etapas por las cuales pasó el grupo.

12.9.1. Oportunidades de Mejora

- Definir y acordar con cliente pruebas BETA y/o de aceptación cuanto antes.
- Intentar hacer que la empresa revise decisión de solo probar la aplicación de forma interna, explicando los riesgos que ello conlleva y las limitantes a la hora de probar.
- Mostrar más datos en presentación, ya tenemos información de defectos y esfuerzo para analizar.
- Hicimos métricas con puntos de función (lo cual no es habitual). Mapear esto con esfuerzo y sacar conclusiones.
- Evaluar datos, métricas y resultados (concretos y tangibles) al finalizar evoluciones.
- Partir un poco más la funcionalidad dentro de cada entregable, que no sea por ejemplo un solo módulo Inspectores, y tratar de sub dividir y gestionar evoluciones más cortas que incluyan pruebas internas y externas en cada subconjunto de funcionalidad.
- Hacer una versión de la app. solo con la funcionalidad de “control de servicio” e intentar probarlo. Así poder probar cosas como: si las palabras que estamos usando son del glosario de le empresa.
- Formatear los documentos de forma que cuando los entreguemos a la empresa puedan ser entendidos.

- Validar todos los documentos a entregar al cliente, y no esperar al final del proyecto para hacerlo.
- Trabajar muy cerca de la empresa, aumentar nuestra presencia e interacción para poder disminuir las barreras de comunicación y que cuando llegue el momento de hacer pruebas beta o la transición, estén las cosas claras, y por otro lado también poder disminuir riesgo de disponibilidad de los stakeholders.
- Da la sensación por el de análisis de comunicación y requerimientos que se relevó y validó de la empresa, estaríamos usando un ciclo de vida incremental más que un evolutivo. Estaríamos haciendo incrementos de desarrollo sobre el primer análisis. Se recomienda re considerar ciclo de vida.

12.9.2. Acciones a Realizar

#	Descripción de la Acción	Fecha estimada
1	Reuniones para acordar con cliente las pruebas a realizar y re evaluar la utilización de la aplicación de forma externa a la empresa.	Previo a finalizar algún entregable.
2	Evaluar métricas y resultados al finalizar cada evolución.	17/10/2015
3	Analizar la división de las funcionalidades para las iteraciones.	10/09/2015
4	Validar documento de Diseño con el cliente	11/09/2015
5	Analizar y reconsiderar el ciclo de vida elegido por el grupo.	10/09/2015
6	Asistir de forma más frecuente a las oficinas del cliente para trabajar algunos días al mes desde ahí.	Al comenzar desarrollo del <i>backend</i> .

Tabla 12-49 acciones a realizar segunda revisión

12.10. Anexo 10 - Informe Tercera Revisión

Grupo:	Proyecto para C.U.T.C.S.A. Andrés Amegeiras Diego Barcia Pedro Trecca	Fecha:	30/11/2015
Carrera:	Ingeniería en Sistemas	Revisores:	Amalia Álvarez Marcelo Cagnani

12.10.1. Fortalezas del Grupo

- Fuerte uso de documentación a lo largo del proyecto.
- Utilización de puntos de función como método de estimación para el desarrollo de proyecto.
- Aplicación desarrollada a la fecha cuenta con interfaz de usuario amigable e intuitivo, y las funcionalidades implementadas son correctas.

12.10.2. Oportunidades de Mejora

Referentes a la presentación:

- No queda del todo claro cuál es el problema a resolver, recién tiene una idea cuando se hablan de los distintos módulos de la aplicación.
- No mostrar organigrama de las personas que entrevistamos dentro de la empresa antes de explicar el fin de la aplicación.
- Cambiar el término “iteración” por “evolución” ya que confunde dado que el ciclo de vida elegido es el evolutivo.
- Mostrar y explicar los requerimientos no funcionales.
- Aclarar si los prototipos son descartables o no, sus objetivos y con quien en la empresa fueron validados.
- Explicar mejor como es la situación de los servicios externos a C.U.T.C.S.A. y como se relacionan entre ellos (Movistar y Handsoft).
- Mostrar métricas, valores, indicadores y esfuerzo en horas del proyecto.
- Analizar si los supuestas de estimación de tiempo con los supuestos de dedicación se cumplieron o no.

Otras:

- No le parece que tenemos suficientes argumentos para justificar un proceso de desarrollo con base tradicional y enfoque ágil (cree que es solamente Tradicional).
- Realizar informe de las pruebas que automatizamos y las que no, explicando los motivos de por qué se tomó un camino u otro. Luego entregar dicho informe a C.U.T.C.S.A. (documento de *testing*).
- Solicitar a C.U.T.C.S.A. para hacer pruebas de la aplicación en el ambiente donde se va a utilizar la aplicación, para poder ver como juega la realidad con las restricciones que tenemos en nuestra aplicación, crear escenarios y ejecutarlos (Si es posible).

12.10.3. Acciones a Realizar

#	Descripción de la Acción	Fecha estimada
1	Re estructuración del orden de los temas a tratar en la presentación.	Previo a presentación final.
2	Agregar información faltante requerida en presentación.	Previo a presentación final.
3	Evaluar métricas y supuestas de dedicación.	15/02/2016
4	Analizar y re considerar el nombre del proceso de desarrollo elegido por el grupo.	05/02/2016
5	Realizar informe de pruebas.	05/02/2016
6	Reunión con referente en empresa solicitando realizar pruebas en ambiente real de uso de la aplicación.	03/02/2016

Tabla 12-50 acciones a realizar tercera revisión

12.11. Anexo 11 - Evolución de la Interfaz de Usuario

12.11.1. Resumen ejecutivo

A partir de la reunión realizada con Daniel Mordeki para validar la interfaz de usuario de la aplicación móvil desarrollada para el proyecto de fin de carrera de Ingeniería en Sistemas en la Universidad ORT Uruguay, se obtuvieron varias sugerencias de cambios y correcciones. La gran mayoría de ellas estaban relacionadas a la alineación de componentes e información, y cómo mejorar la presentación. Todas ellas se pueden apreciar en los resultados de la reunión, así como las mejoras realizadas a la aplicación luego de la misma, teniendo en cuenta los puntos mencionados.

12.11.2. Objetivo

Presentar los resultados de la reunión con Daniel Mordeki, quien tiene un alto nivel de experiencia en interfaz de usuario y usabilidad, con el fin de validar y mejorar la interfaz de usuario de la aplicación en desarrollo hasta el momento. Dicha reunión se realizó el 5 de octubre del 2015.

A su vez, mostrar cómo evolucionó la aplicación antes y después de la reunión, indicando las correcciones realizadas en función de los puntos marcados en la reunión.

12.11.3. Reunión

Fecha	5 de octubre del 2015
Participantes	Diego Barcia, Daniel Mordeki.
Duración	80 minutos.
Elementos de entrada	Aplicación móvil con interfaz de usuario desarrollada

Tabla 12-51 Resumen de Reunion

En la reunión con Daniel Mordeki, se le presentó un flujo completo de la aplicación, pasando por todas las funcionalidades de la misma. A medida que se transcurría por cada pantalla, Daniel revisaba la interfaz y proponía cambios o mejoras, las cuales se listan a continuación:

1. Toda imagen o texto no puede quedar nunca pegada al borde de la pantalla

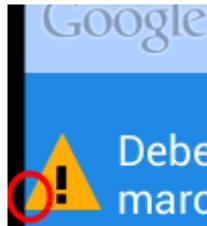


Ilustración 12-80 Imagen pegada al borde

2. No decirle al usuario que haga las cosas. Existen dos alternativas:
 - La aplicación lo hace sola.
 - Crear una interfaz para que haga la acción.
3. La información del marcador debe tapar toda la pantalla, no mostrarse reducida.

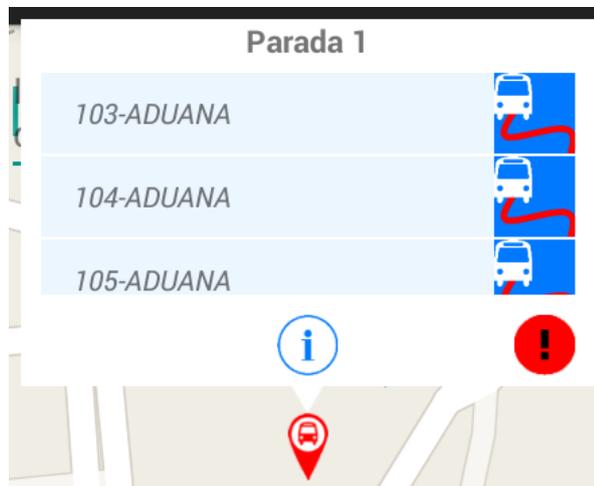


Ilustración 12-81 información parcial en la pantalla

4. El color rojo genera al usuario una sensación de peligro.



Ilustración 12-82 Imagen Sensación de peligro

5. Indicar el sentido de la línea cuando se muestra la información sobre ellas.
6. Agregar mayor sangría en las sublistas. Que se note la jerarquía del padre y sus componentes hijos.



Ilustración 12-83 Sangría en sublistas

- Dejar mayor espacio entre un elemento padre de la lista y el siguiente. Que se note que son elementos distintos.

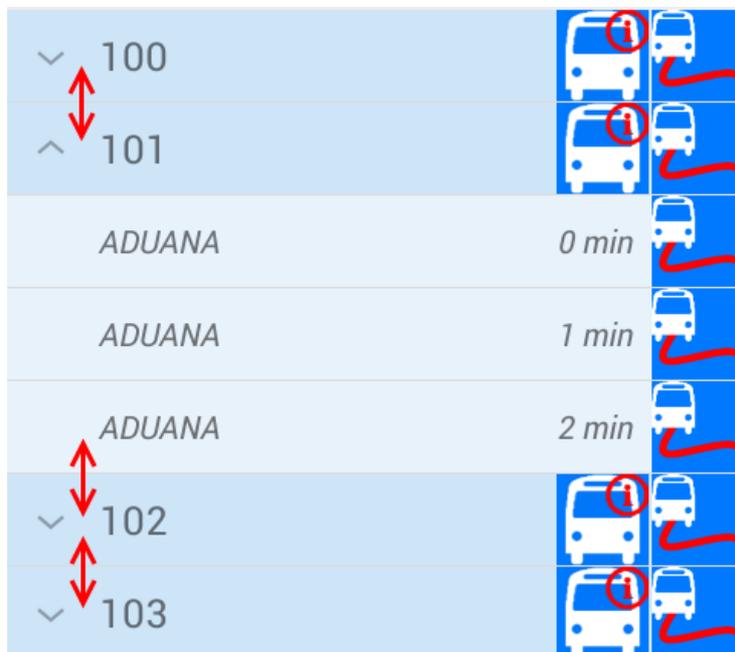


Ilustración 12-84 Sangría entre padres

- El botón con signo de exclamación, así como se muestra, el usuario no conoce su razón ni uso.

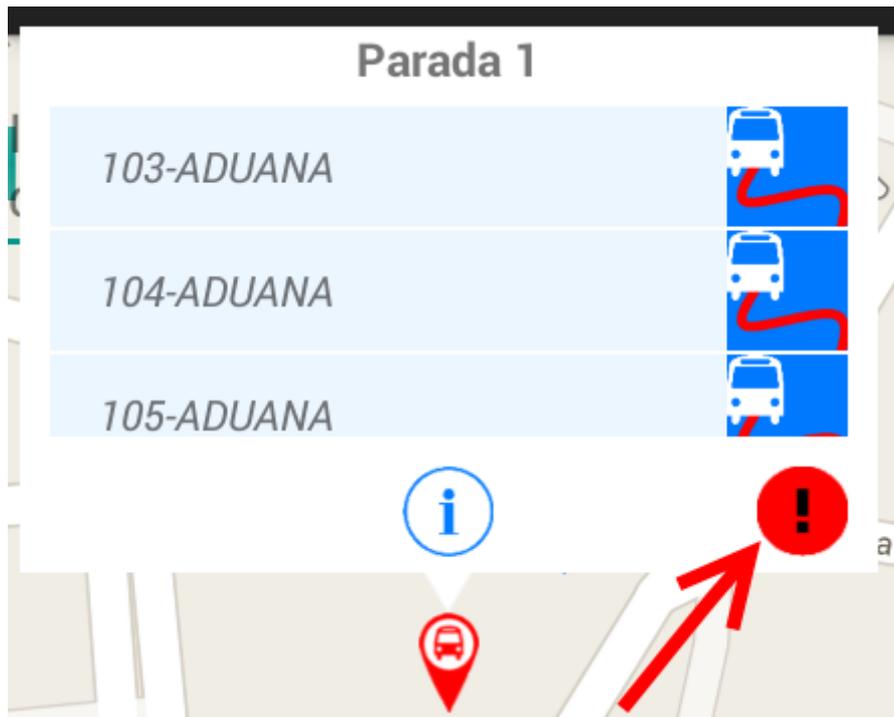


Ilustración 12-85 botón sin uso claro

9. Mostrar información de cuántos minutos faltan para que llegue el próximo ómnibus ya mismo en la información del marcador de la parada.

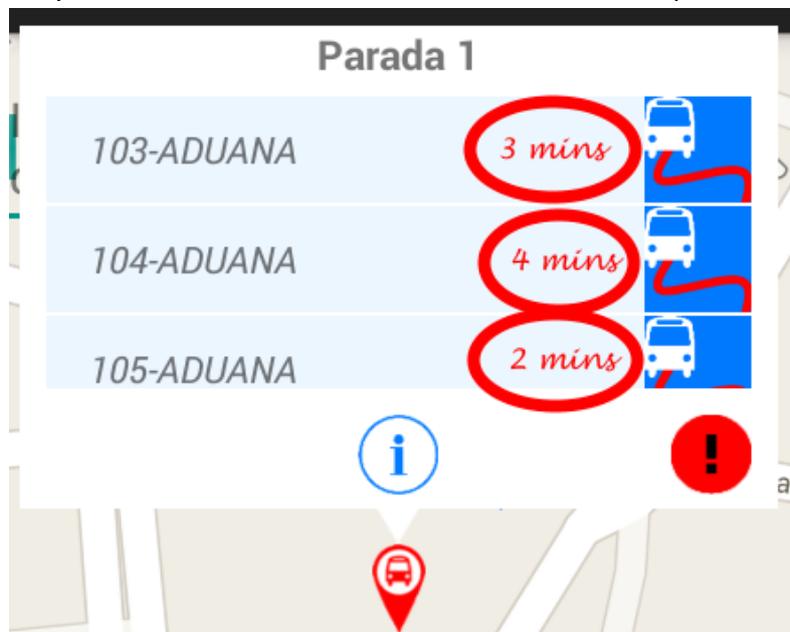


Ilustración 12-86 Falta de información

10. Agregar animación para transiciones.
11. Rediseñar el *popup* del marcador, actualizando a cómo se usan actualmente (por ejemplo, Google Maps para Android).

12. Separar las etiquetas y datos y ordenarlas para que se muestren de forma más simétrica, no todo tirado hacia la izquierda.

Nro Conexión: 3130
Salida planificada: 08:15
Llegada planificada: 09:07
SAM: 60096-EOAEF3C920B93

Ilustración 12-87 Información más simétrica

13. Aumentar el tamaño del texto en la información del viaje. Aprovechar el *scroll* para poder sobrepasar la cantidad de elementos que entran en la pantalla.
14. Usar sangría para indicar la jerarquía padre-hijo.

103-ADUANA

↔ Nro Conexión: 3130

↔ Salida planificada: 08:15

↔ Llegada planificada: 09:07

↔ SAM: 60096-EOAEF3C920B93

Ilustración 12-88 Sangría para jerarquía Padre-Hijo

15. Las imágenes no deben quedar recortadas.

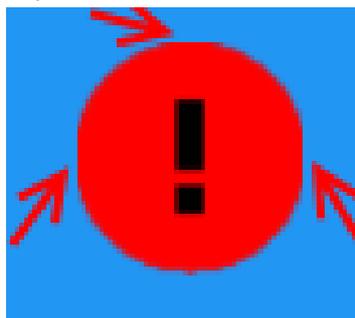


Ilustración 12-89 Imagen Recortada

12.11.4. Las 3 reglas de oro

Al finalizar la reunión, Daniel propuso 3 reglas básicas que se deben seguir para tener una interfaz de usuario aceptable:

1. **Alineación:** mantener los distintos componentes alineados tanto horizontal como verticalmente. De lo contrario el ojo lo detecta como desprolijo.
2. **Empujar contenido y funcionalidad hacia arriba:** Siempre que se pueda, mostrarle la información al usuario lo antes posible. Cuantos menos *clicks* deba ejecutar, mejor.
3. **Agrupación visual (representar jerarquías de familias):** Distinguir los componentes hijos, padres y hermanos. Que se note qué componente se encuentra incluido en otro, y qué componentes son estructuralmente similares entre sí (pero distintos en contenido).

12.11.5. Correcciones y evolución

Participantes	Diego Barcia
Elementos de entrada	<ul style="list-style-type: none">● Aplicación móvil con interfaz de usuario desarrollada● Notas de reunión con Daniel Mordeki● Página web Material Designⁱ

Tabla 12-52 Responsables de corrección

A continuación, se muestra el antes y el después del diseño de la interfaz de usuario a partir de las correcciones y recomendaciones indicadas por Daniel Mordeki en la reunión, y las recomendaciones que ofrece Google en su página de *Material Design*.

Login	
Antes	Después
 <p style="text-align: center;"><i>Ilustración 12-90 Log in Antes</i></p>	 <p style="text-align: center;"><i>Ilustración 12-91 Log in después</i></p>
<ul style="list-style-type: none"> • Se modifica el botón de INICIAR SESIÓN según las recomendaciones de Google. 	

Tabla 12-53 Log In

Vista de error al ingresar datos erróneos en el <i>login</i>	
Antes	Después
 <p data-bbox="247 1332 542 1366"><i>Ilustración 12-92 Log in Error</i></p>	 <p data-bbox="766 1321 1149 1355"><i>Ilustración 12-93 Log in Error Después</i></p>
No hubo cambios.	

Tabla 12-54 Vista de error al ingresar datos erróneos en el login

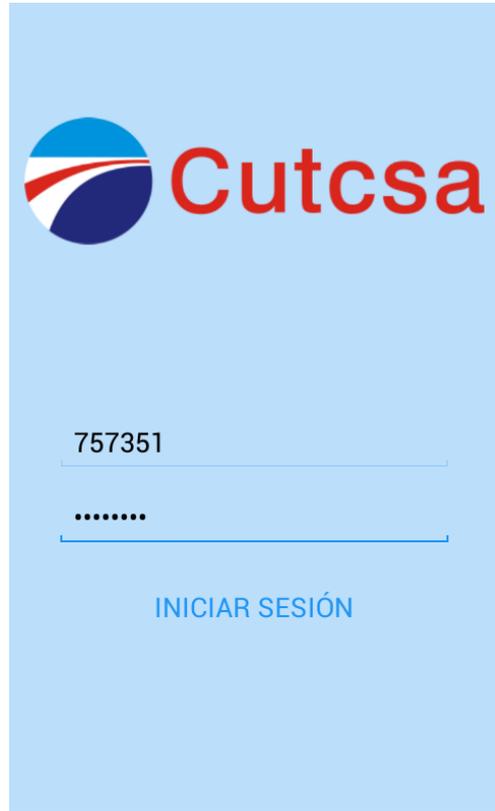
Ingreso de datos del <i>login</i>	
Antes	Después
 <p><i>Ilustración 12-94 Log in cambio de texto antes</i></p>	 <p><i>Ilustración 12-95 Log in cambio de texto</i></p>
<ul style="list-style-type: none"> El color blanco de los datos a llenar no tenía suficiente contraste con el fondo de la aplicación, por lo que se cambia a negro. 	

Tabla 12-55 Ingreso de datos del login

Vista del mapa que mostraría las paradas de Montevideo y cercanías

Antes



Ilustración 12-96 Mapa antes del cambio

Después

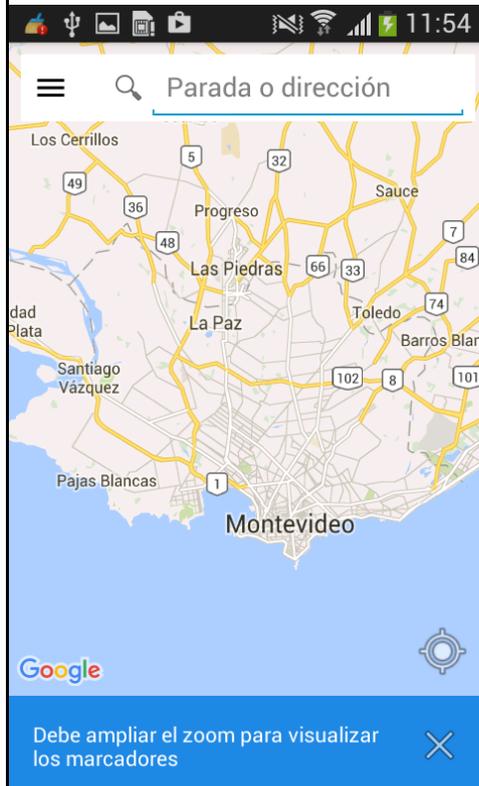
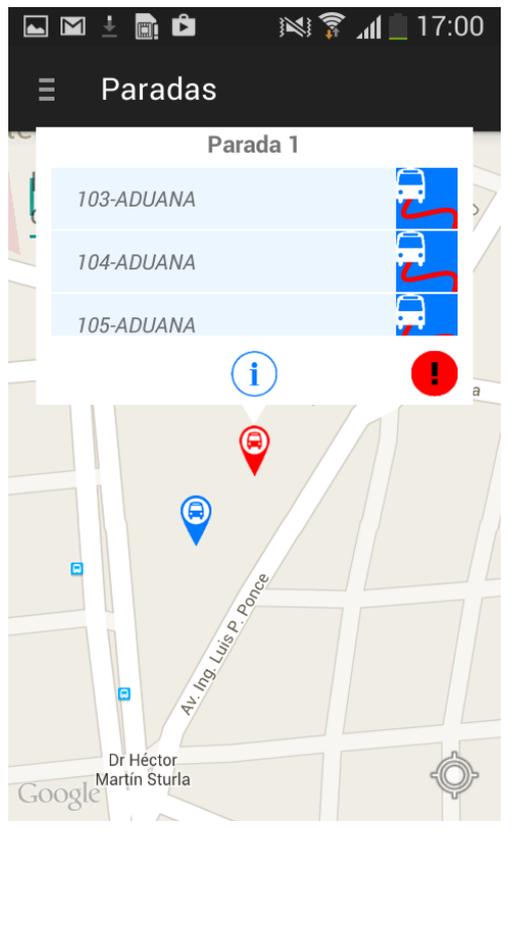
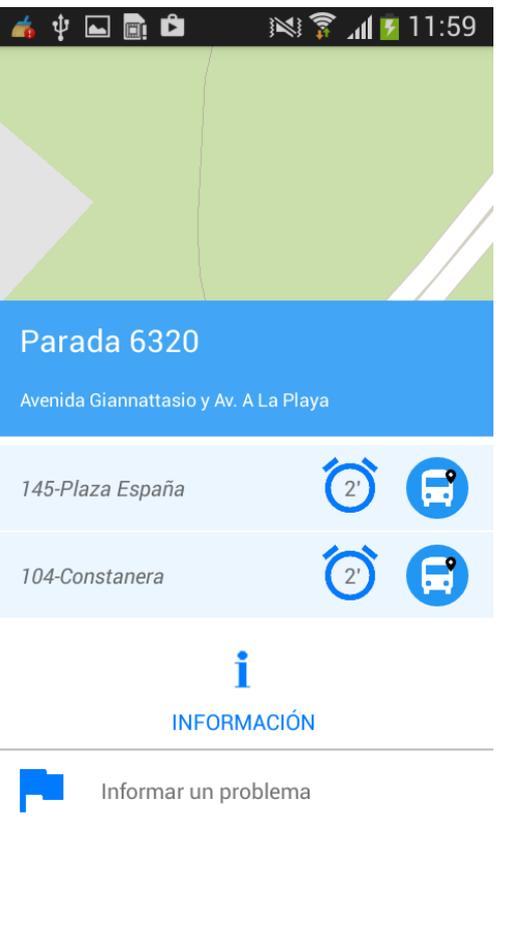


Ilustración 12-97 Mapa después del cambio

- Se quita el *action* bar (barra superior donde se puede acceder al menú lateral), para reemplazar la accesibilidad al menú de la misma forma que lo hace Google en su aplicación Mapa para Android. De esta forma se gana espacio de visualización del mapa.
- Se quita la imagen de advertencia del mensaje inferior de la pantalla, y se aumenta el tamaño del botón de cerrar dicho mensaje. También se centra el texto y los botones en el componente del mensaje, y se le agrega un *padding* para que no quede muy pegado a los márgenes.

Tabla 12-56 Ingreso de datos del login

Información del marcador de una parada

Antes	Después
 <p data-bbox="240 1288 778 1377"><i>Ilustración 12-98 Información de una parada antes del cambio</i></p>	 <p data-bbox="778 1288 1316 1377"><i>Ilustración 12-99 Información de una parada después del cambio</i></p>

- El *popup* del marcador tuvo un cambio radical, siguiendo la forma en que Google muestra la información de los marcadores en su aplicación Mapa.
- Se cambia la imagen del traqueo de ómnibus  por una más representativa y con un nuevo diseño .
- Se explicita el botón de información  mostrándolo de la forma en que lo hace y recomienda Google.
- El botón de reporte de error , el cual no quedaba muy explícito, se muestra de la forma en que Google usa para que el usuario reporte un problema.
- Se agrega un efecto de despliegue al seleccionar y deseleccionar un marcador, para que el mostrar y ocultar la información del marcador (y el

componente de búsqueda del mapa) sea más animado (similar al que se usa en la aplicación Mapas de Google).

- Se “trae información hacia arriba”, permitiendo ver en esta vista el primer ómnibus que se acerca a la parada por cada recorrido. Esto se muestra primero con una indicación de carga (mientras se obtiene la información del servidor) y se muestra al usuario indicando cuántos minutos faltan para

que llegue el ómnibus: de  a .

- Toda información que debe ser cargada, tanto en esta vista como en otras a lo largo de la aplicación móvil, tiene una notificación hacia el usuario por medio de sonido (una pequeña alarma) y una breve vibración del dispositivo para avisarle que la información terminó de cargar. Dichas notificaciones no existían en la versión de la aplicación presentada a Daniel Mordeki en la reunión. Las mismas fueron agregadas a raíz de que los usuarios estarán la mayoría del tiempo utilizando la aplicación en la calle o sobre el ómnibus, y no pueden estar constantemente mirando el dispositivo para saber si terminó de cargar la información. Una pequeña alerta mediante sonido y/o vibración, permite al usuario prestar atención en otros elementos importantes de su contexto (por ejemplo, ver si el ómnibus ya se está acercando a la parada), y continuar el uso de la aplicación cuando la carga realmente finalizó.
- Se permite minimizar la información del marcador, mostrando así la información de cabecera (número de la parada y su dirección) y un acceso directo a la función de “cómo ir en ómnibus”:

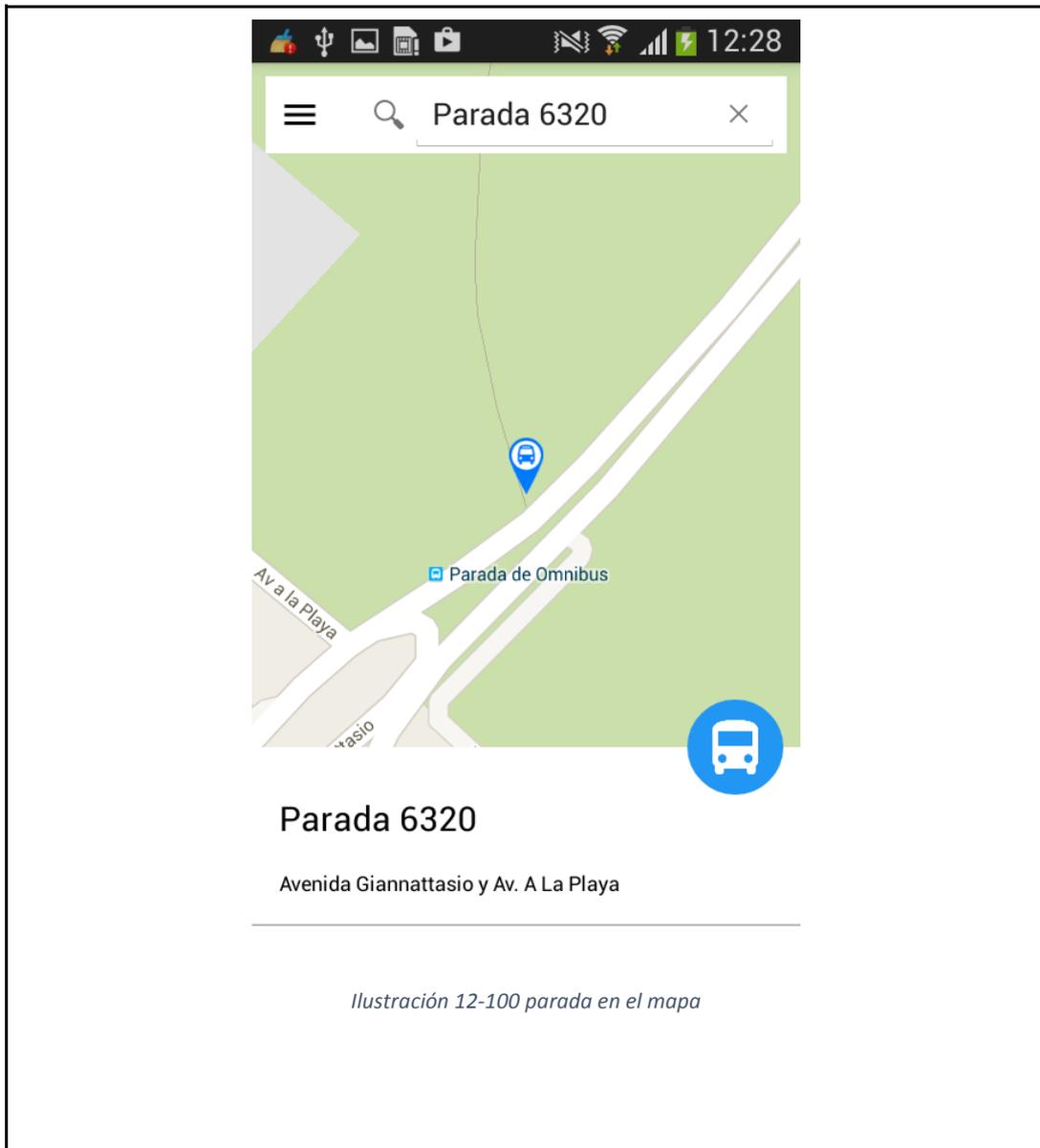


Ilustración 12-100 parada en el mapa

Tabla 12-57 Información del marcador de una parada

Información detallada de una parada

Antes

Después

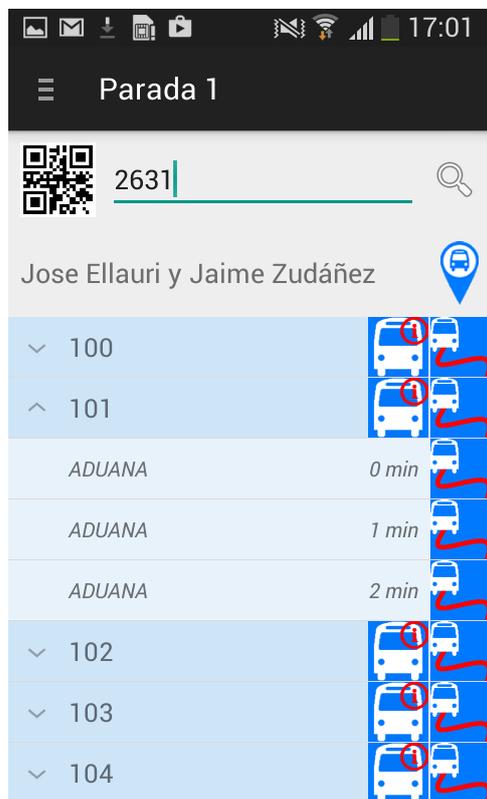


Ilustración 12-101 Información de una parada antes del cambio

Ilustración 12-102 Información de una parada después del cambio

- Teniendo en cuenta la recomendación de hacer más visible la jerarquía padre/hijo/hermano, se agrega más sangría a los elementos internos de la lista desplegable.
- Se agrega la información del sentido (dado que puede pasar que en una misma parada pase un recorrido en ambos sentidos).
- Se modifican los colores para que sean más representativos y sean coherentes con el color de la aplicación.

- Las imágenes de información de la línea  y ver el ómnibus en el mapa  se modificaron hacia una forma que se entienda mejor y con un

nuevo diseño:  y  respectivamente.

- Mientras se van obteniendo la información de los próximos arribos, se agrega una notificación de estado al usuario:



Ilustración 12-103 Cargando información de arribo

Tabla 12-58 Información detallada de una parada

Vista del mapa que muestra un recorrido

Antes

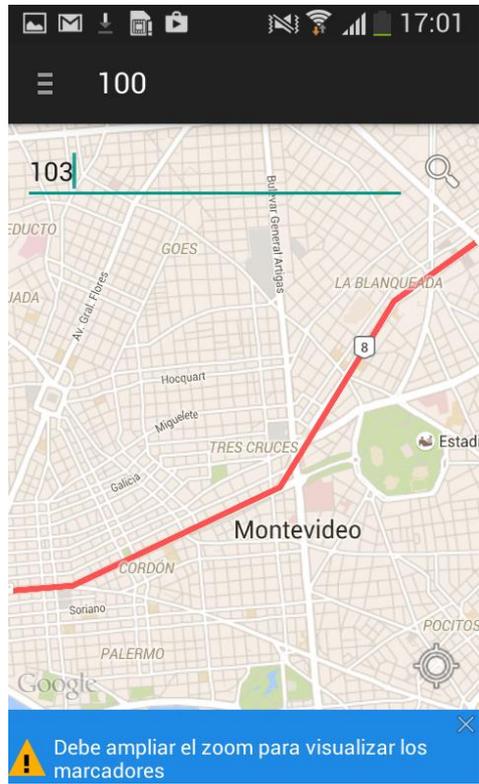


Ilustración 12-104 Camino de un coche antes

Después

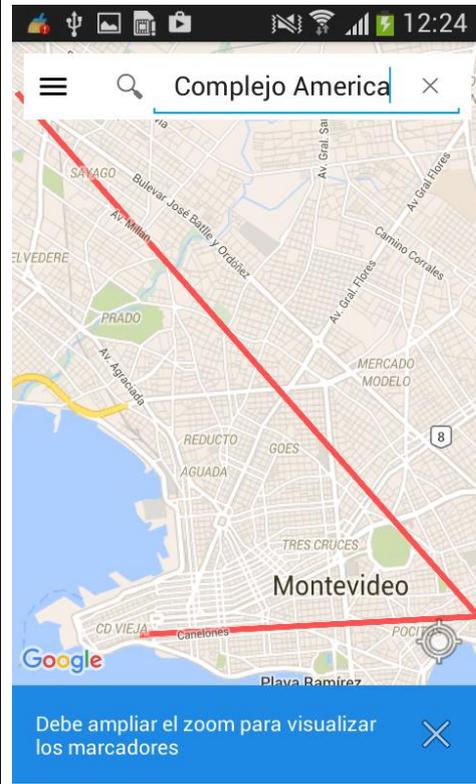


Ilustración 12-105 Camino de un coche después

No hubo cambios en esta vista más que los mencionados anteriormente.

Tabla 12-59 Vista del mapa que muestra un recorrido

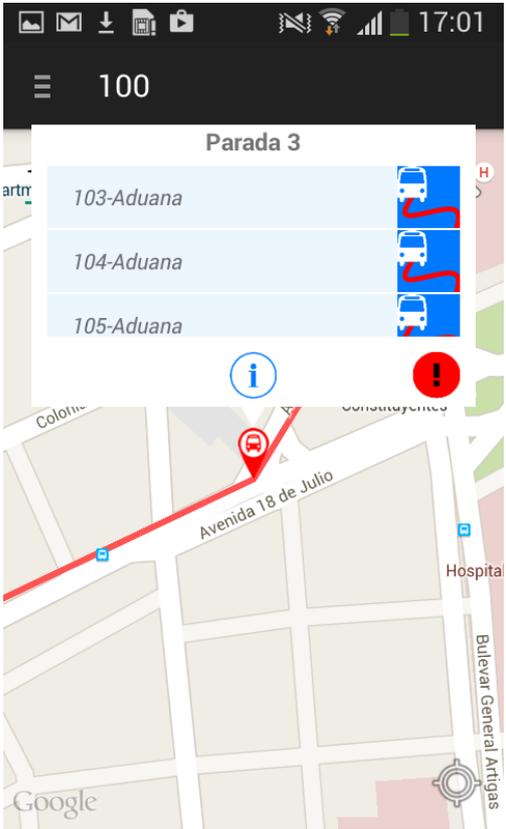
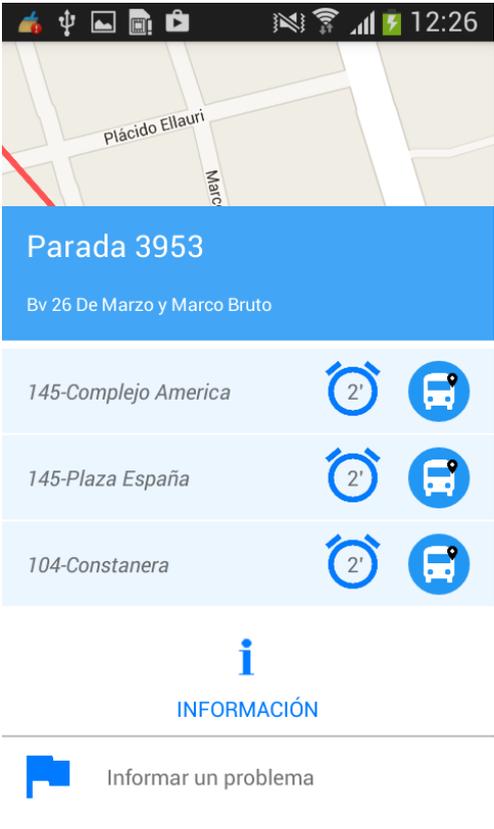
Información de un marcador de la parada por donde pasa un recorrido	
Antes	Después
 <p><i>Ilustración 12-106 Información de una parada antes del cambio</i></p>	 <p><i>Ilustración 12-107 Información de una parada después del cambio</i></p>
<p>Los cambios representados aquí se indicaron anteriormente en la información del marcador de la parada.</p>	

Tabla 12-60 Información de un marcador de la parada por donde pasa un recorrido

Vista del mapa con coches

Antes



Ilustración 12-108 Coche en el mapa antes del cambio

Después

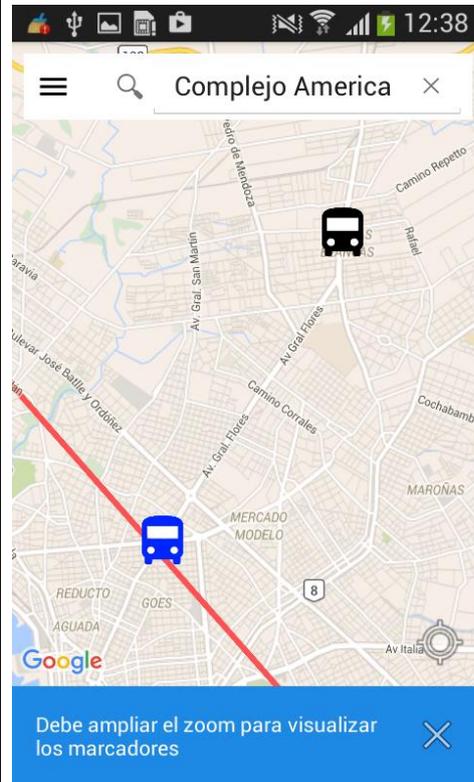


Ilustración 12-109 Coche en el mapa después del cambio

- Para ayudar a la accesibilidad por personas con problemas de visión cromática, se indica en el mapa con color azul el coche del que se está obteniendo la posición del GPS, y negro cuando no se tiene dicha información.
- Se cambian las imágenes de los ómnibus a los ofrecidos por Google en Material Design.

Tabla 12-61 Vista del mapa con coches

Información del marcador de un coche

Antes

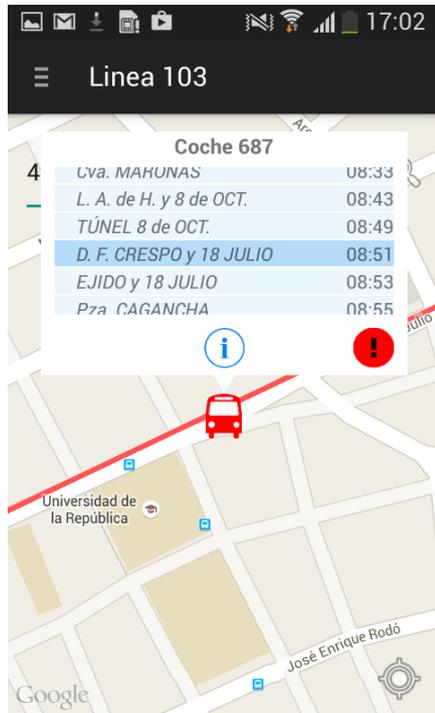


Ilustración 12-110 Información del Coche antes del cambio

Después



Ilustración 12-111 Información del Coche después del cambio

- Al seleccionar un coche, se muestra la información básica del mismo. Dicha forma de mostrar la información se modificó a una forma similar a la que se usa con los marcadores de las paradas.
- En caso de seleccionar un coche cuya posición de GPS no se conoce, se agrega la información al respecto.
- Se puede minimizar la información del marcador mostrando sólo la información del número de coche:

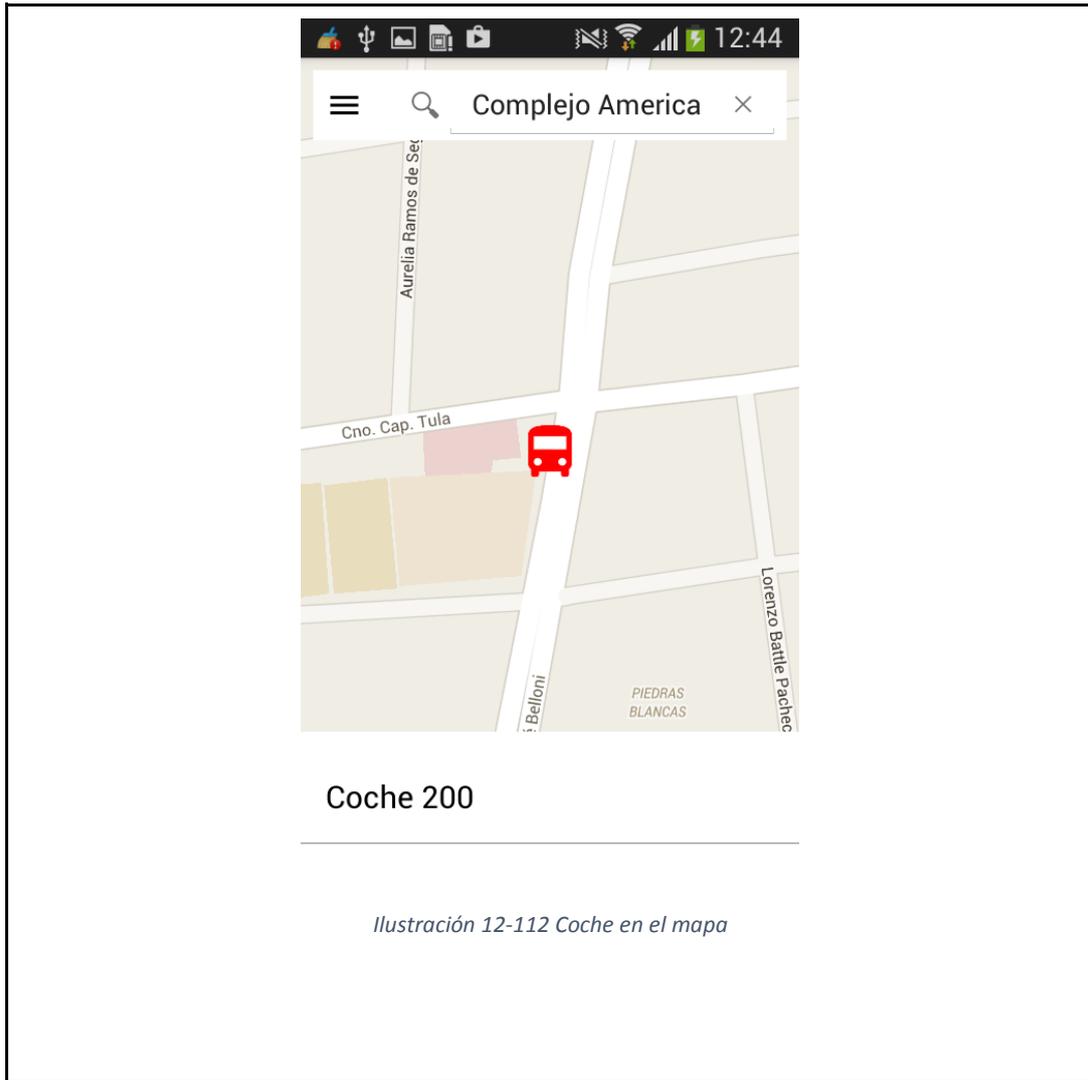


Ilustración 12-112 Coche en el mapa

Tabla 12-62 Información del marcador de un coche

Ver información básica de un servicio de un coche

Antes

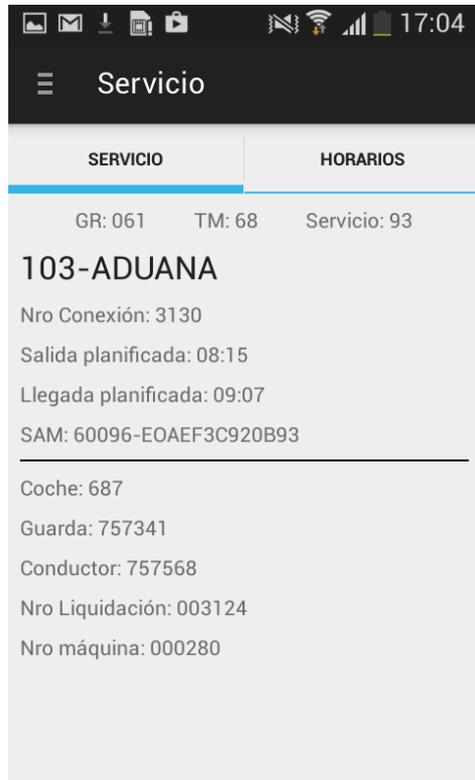


Ilustración 12-113 Información del servicio antes del cambio

Después

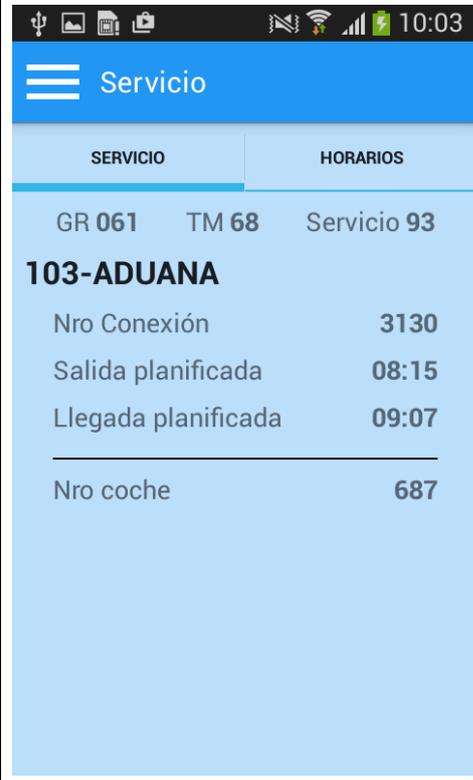


Ilustración 12-114 Información del servicio después del cambio

- Se cambia el color del fondo para que sea acorde al color de la aplicación.
- Se cambia el tamaño de la letra para que sea más legible, y ayude a entender la jerarquía de los elementos: Nro Conexión, Salida planificada, Llegada planificada, Nro coche son hijos (pertenecen a) 104-Geant.
- Para la jerarquía también se agregó la sangría para dejar más claro qué información pertenece a qué componente.
- Se usa el estilo "negrita" para resaltar la información sobre los *labels*, ayudando al usuario a enfocar su atención en la información en sí.
- Hay información que fue retirada para la nueva versión de la aplicación, dado que era información que no se debía mostrar en esta vista.

Tabla 12-63 Ver información básica de un servicio de un coche

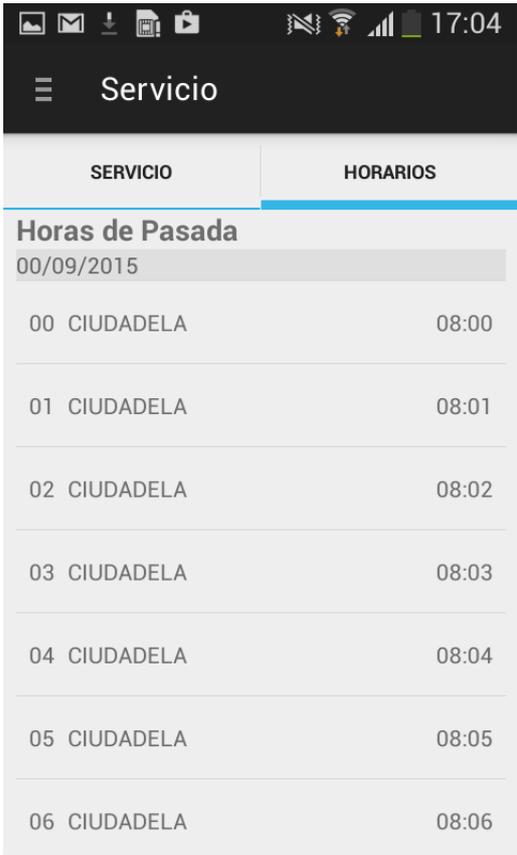
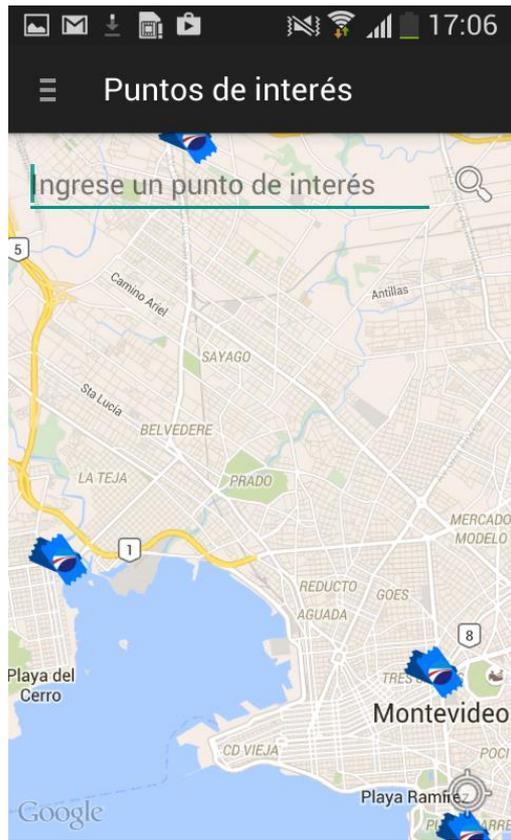
Información de horarios de pasada por puntos de control de un servicio de un coche	
Antes	Después
 <p><i>Ilustración 12-115 Ver Horarios antes del cambio</i></p>	 <p><i>Ilustración 12-116 Ver Horarios después del cambio</i></p>
<ul style="list-style-type: none"> ● Se cambia el color del fondo para que sea acorde al color de la aplicación. ● La lista se despliega hasta los bordes derecho e izquierdo del dispositivo. A quienes se les coloca margen es a los elementos de la lista. 	

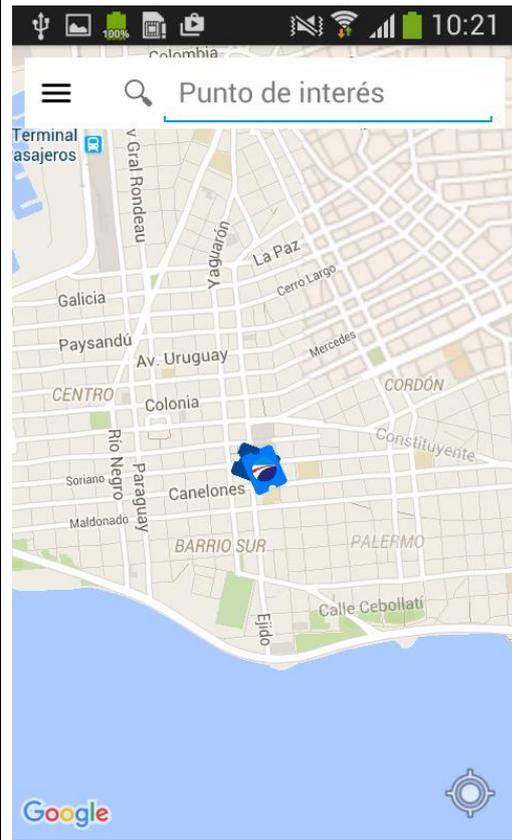
Tabla 12-64 Información de horarios de pasada por puntos de control de un servicio de un coche

Vista del mapa mostrando puntos de interés

Antes



Después

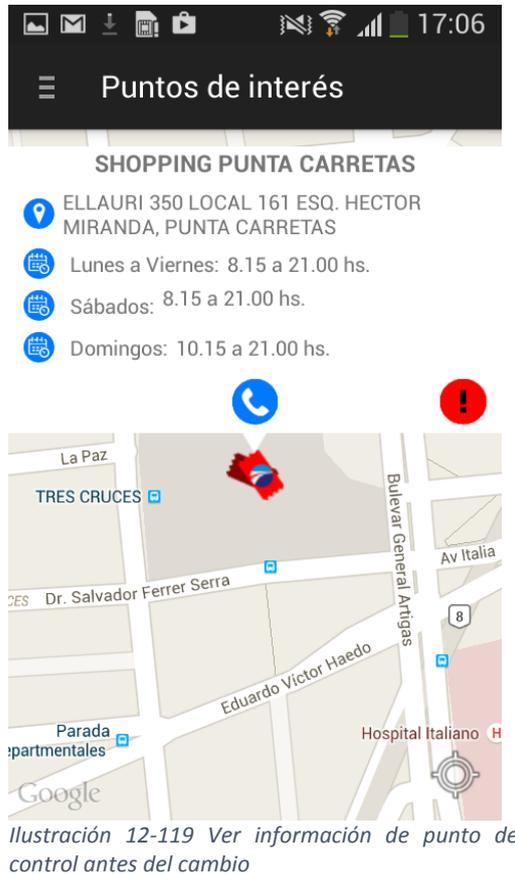


- Se cambia la imagen del ticket a la ofrecida por Google en Material Design.

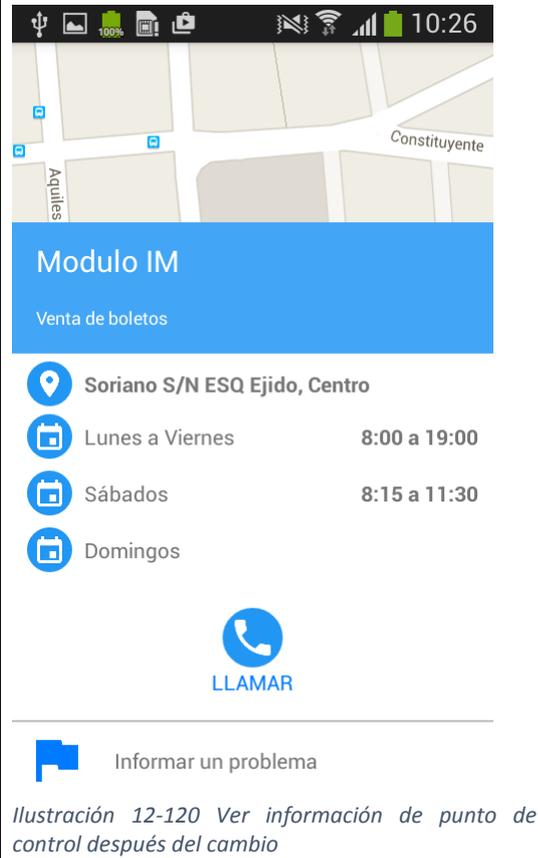
Tabla 12-65 Vista del mapa mostrando puntos de interés

Información de puntos de interés

Antes



Después



- Al seleccionar un punto de interés, se muestra la información básica del mismo. Dicha forma de mostrar la información se modificó a una forma similar a la que se usa con los marcadores de las paradas.
- Se cambian las imágenes a los ofrecidos por Google en su página de Material Design.
- Se corrige alineación de los horarios del sábado.
- Se alinea la información para que aparezcan a la misma altura.
- Se resalta con el estilo “negrita” la información para que se destaque frente a los *labels*.
- La opción de LLAMAR se explicita siguiendo las recomendaciones de Google en Material Design.
- Al minimizar la información del marcador, muestra el nombre del punto de interés y su tipo (por ejemplo, Venta de boletos), y un acceso directo a la función de “cómo ir en ómnibus”:

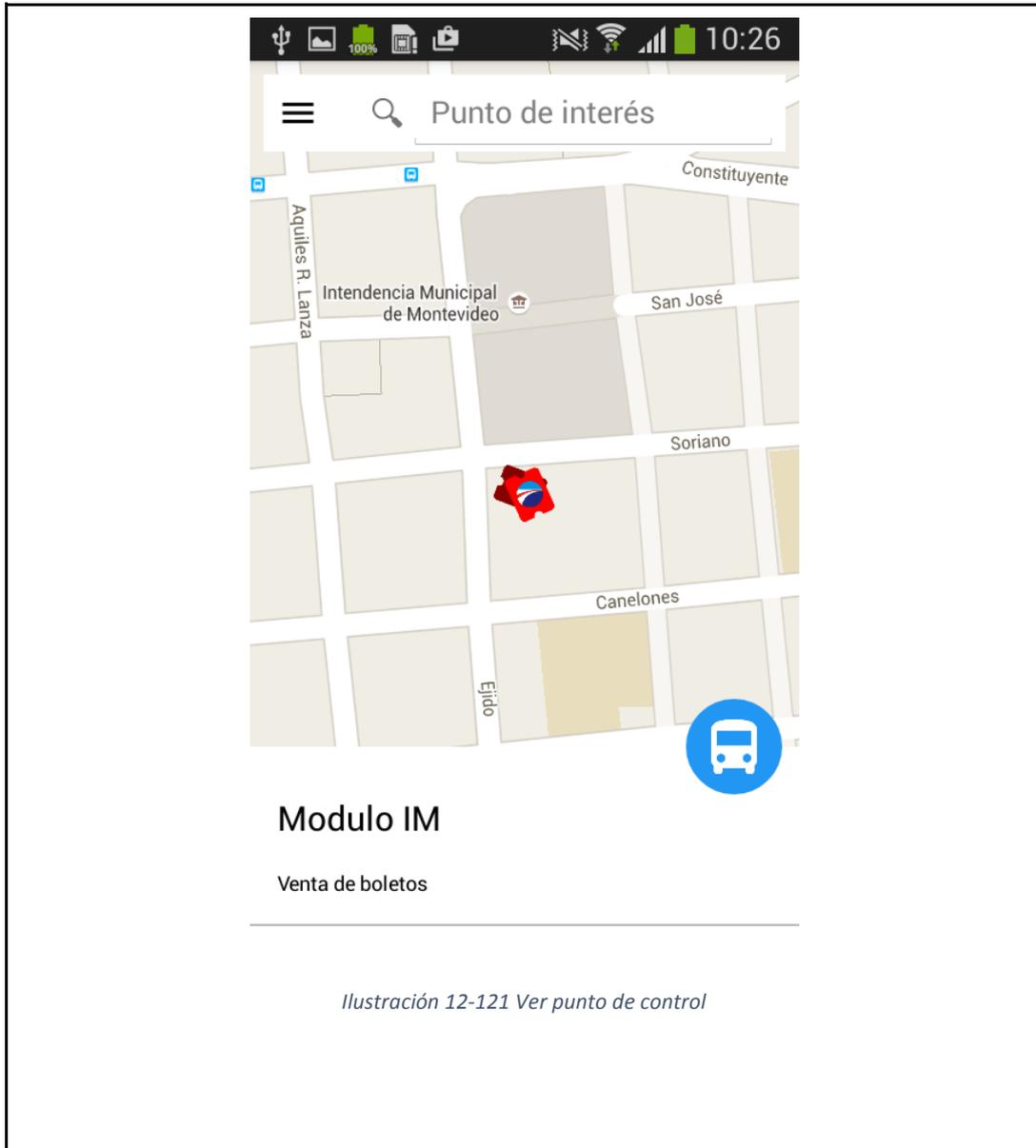


Ilustración 12-121 Ver punto de control

Tabla 12-66 Información de puntos de interés

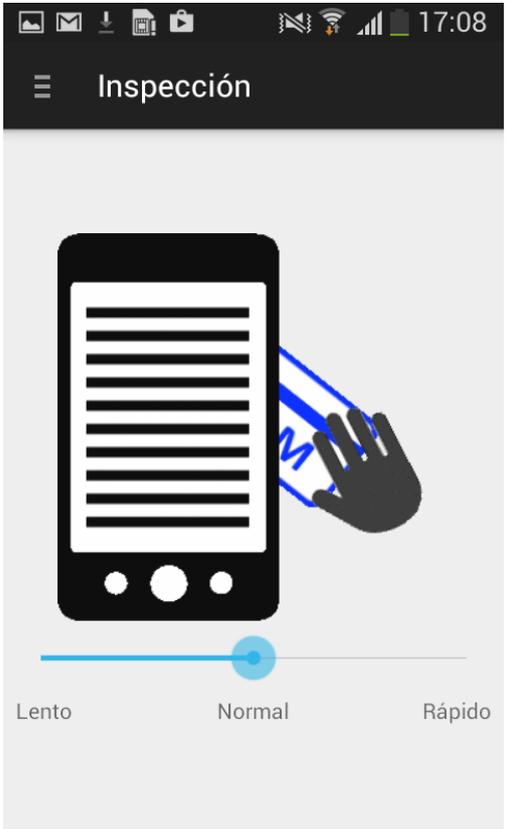
Tutorial para el control de un viaje	
Antes	Después
 <p><i>Ilustración 12-122 tutorial antes del cambio</i></p>	 <p><i>Ilustración 12-123 tutorial después del cambio</i></p>
<ul style="list-style-type: none"> ● Se cambia el color del fondo para que sea acorde al color de la aplicación. ● Se agrega una descripción por texto en cada paso del tutorial. ● Se agregan controles de reproducción para pausar la reproducción del GIF, pasar a la siguiente imagen, y volver a la imagen anterior. ● El control de reproducción desaparece automáticamente después de breves segundos si el usuario presiona sobre ningún botón del control (pausa, play, reproducir imagen siguiente, reproducir imagen anterior). ● Cuando el control de reproducción no se encuentra visible, si el usuario presiona sobre la imagen, vuelve a aparecer. 	

Tabla 12-67 Tutorial para el control de un viaje

Control de información básica de un viaje	
Antes	Después
 <p>16/09/2015 08:00 CONTROLAR</p> <p>SERVICIO ANULADOS HORARIOS</p> <p>GR: 061 TM: 68 Servicio: 93</p> <p>103-ADUANA</p> <p>Nro Conexión: 3130 Salida planificada: 08:15 Llegada planificada: 09:07 SAM: 60096-EOAEF3C920B93</p> <hr/> <p>Coche: 687 Guarda: 757341 Conductor: 757568 Nro Liquidación: 003124 Nro máquina: 000280</p> <p><i>Ilustración 12-124 Vista información de viaje de servicios antes del cambio</i></p>	 <p>20/11/2015 11:17:55 CONTROLAR</p> <p>SERVICIO ANULADOS HORARIOS</p> <p>GR 061 TM 3 Servicio 163</p> <p>145-Complejo America</p> <p>Nro Conexión 3130 Salida planificada 08:15 Llegada planificada 09:07</p> <p>60096-EOAEF3C920B93</p> <hr/> <p>Nro coche 414 Conductor 34453893 Nro liquidación 003194</p> <p><i>Ilustración 12-125 vista información de viaje después del cambio</i></p>

- Se modifica el botón CONTROLAR siguiendo las recomendaciones de diseño que ofrece Google en Material Design.
- Se cambia el color del fondo para que sea acorde al color de la aplicación.
- Se cambia el tamaño de la letra para que sea más legible, y ayude a entender la jerarquía de los elementos. Se aprovecha el *scroll* para los casos en que la información no quepa en la vista.
- Para la jerarquía también se agregó la sangría para dejar más claro qué información pertenece a qué componente.
- Se usa el estilo “negrita” para resaltar la información sobre los *labels*, ayudando al usuario a enfocar su atención en la información en sí.

Tabla 12-68 Control de información básica de un viaje

Control de boletos anulados de un viaje																																																							
Antes	Después																																																						
 <p>16/09/2015 08:00</p> <p>CONTROLAR</p> <table border="1"> <thead> <tr> <th>SERVICIO</th> <th>ANULADOS</th> <th>HORARIOS</th> </tr> </thead> <tbody> <tr> <td colspan="3">Total boletos anulados: 7</td> </tr> <tr> <td colspan="3">Boletos Anulados</td> </tr> <tr> <td colspan="3">00/09/2015</td> </tr> <tr> <td>0000006810</td> <td></td> <td>10:00 \$26</td> </tr> <tr> <td>0000006811</td> <td></td> <td>10:01 \$26</td> </tr> <tr> <td>0000006812</td> <td></td> <td>10:02 \$26</td> </tr> <tr> <td>0000006813</td> <td></td> <td>10:03 \$26</td> </tr> <tr> <td>0000006814</td> <td></td> <td>10:04 \$26</td> </tr> </tbody> </table> <p><i>Ilustración 12-126 Vista información de viaje de servicios antes del cambio</i></p>	SERVICIO	ANULADOS	HORARIOS	Total boletos anulados: 7			Boletos Anulados			00/09/2015			0000006810		10:00 \$26	0000006811		10:01 \$26	0000006812		10:02 \$26	0000006813		10:03 \$26	0000006814		10:04 \$26	 <p>20/11/2015 11:09:23</p> <p>CONTROLAR</p> <table border="1"> <thead> <tr> <th>SERVICIO</th> <th>ANULADOS</th> <th>HORARIOS</th> </tr> </thead> <tbody> <tr> <td colspan="3">Total boletos anulados 6</td> </tr> <tr> <td colspan="3">Boletos Anulados</td> </tr> <tr> <td colspan="3">01/09/2015</td> </tr> <tr> <td>0000006810</td> <td></td> <td>10:00 \$26</td> </tr> <tr> <td>0000006815</td> <td></td> <td>10:12 \$13</td> </tr> <tr> <td>0000006823</td> <td></td> <td>10:20 \$13</td> </tr> <tr> <td>0000006878</td> <td></td> <td>10:55 \$26</td> </tr> <tr> <td>0000006889</td> <td></td> <td>10:59 \$13</td> </tr> </tbody> </table> <p><i>Ilustración 12-127 vista información de viaje después del cambio</i></p>	SERVICIO	ANULADOS	HORARIOS	Total boletos anulados 6			Boletos Anulados			01/09/2015			0000006810		10:00 \$26	0000006815		10:12 \$13	0000006823		10:20 \$13	0000006878		10:55 \$26	0000006889		10:59 \$13
SERVICIO	ANULADOS	HORARIOS																																																					
Total boletos anulados: 7																																																							
Boletos Anulados																																																							
00/09/2015																																																							
0000006810		10:00 \$26																																																					
0000006811		10:01 \$26																																																					
0000006812		10:02 \$26																																																					
0000006813		10:03 \$26																																																					
0000006814		10:04 \$26																																																					
SERVICIO	ANULADOS	HORARIOS																																																					
Total boletos anulados 6																																																							
Boletos Anulados																																																							
01/09/2015																																																							
0000006810		10:00 \$26																																																					
0000006815		10:12 \$13																																																					
0000006823		10:20 \$13																																																					
0000006878		10:55 \$26																																																					
0000006889		10:59 \$13																																																					
<ul style="list-style-type: none"> ● Se destaca con estilo “negrita” la información de cantidad de boletos anulados para resaltarlo del <i>label</i>. ● Se cambia el color del fondo para que sea acorde al color de la aplicación. ● Se alinea la lista hasta los bordes de la pantalla del dispositivo, y se les agrega margen a los elementos internos de la lista. 																																																							

Tabla 12-69 Control de boletos anulados de un viaje

Control de horarios de pasada por puntos de control de un viaje

Antes

Después



Ilustración 12-128 Vista información de viaje de servicios antes del cambio

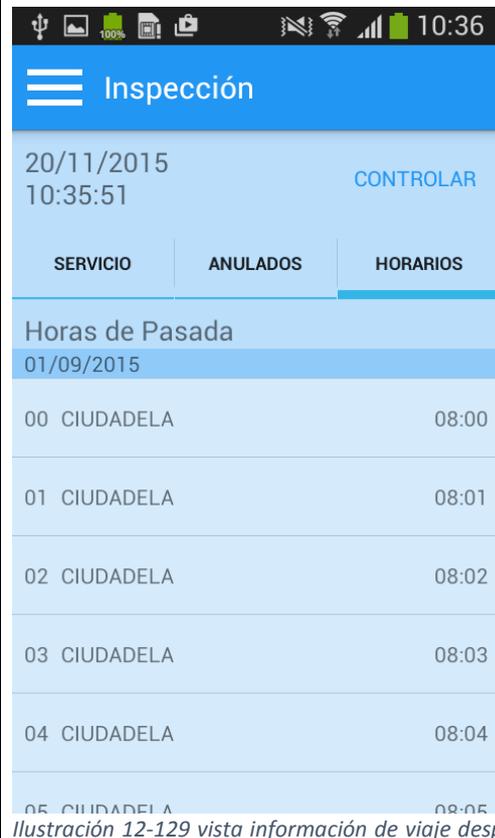


Ilustración 12-129 vista información de viaje después del cambio

- Se cambia el color del fondo para que sea acorde al color de la aplicación.
- La lista se despliega hasta los bordes derecho e izquierdo del dispositivo. A quienes se les coloca margen es a los elementos de la lista.

Tabla 12-70 Control de horarios de pasada por puntos de control de un viaje

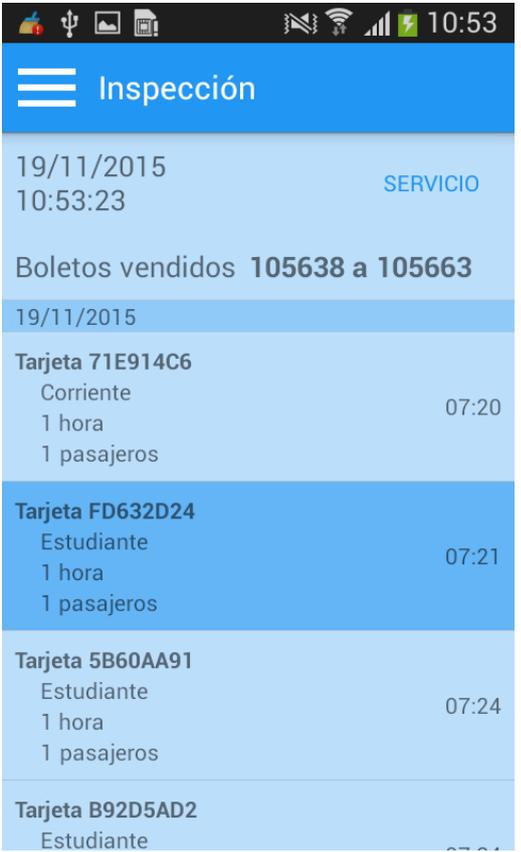
Control de tarjeta de pasajeros de un viaje																																																					
Antes	Después																																																				
 <p>16/09/2015 08:00 SERVICIO</p> <p>Boletos vendidos: 6801 a 7194</p> <p>00/09/2015</p> <table border="1"> <tr> <td>F69B90F0</td> <td>10:00</td> </tr> <tr> <td>Estudiante 1 hora</td> <td></td> </tr> <tr> <td>F69B90F1</td> <td>10:01</td> </tr> <tr> <td>Estudiante 1 hora</td> <td></td> </tr> <tr> <td>F69B90F2</td> <td>10:02</td> </tr> <tr> <td>Estudiante 1 hora</td> <td></td> </tr> <tr> <td>F69B90F3</td> <td>10:03</td> </tr> <tr> <td>Estudiante 1 hora</td> <td></td> </tr> <tr> <td>F69B90F4</td> <td>10:04</td> </tr> <tr> <td>Estudiante 1 hora</td> <td></td> </tr> <tr> <td>F69B90F5</td> <td>10:05</td> </tr> <tr> <td>Estudiante 1 hora</td> <td></td> </tr> </table> <p><i>Ilustración 12-130 Control de boleto antes del cambio</i></p>	F69B90F0	10:00	Estudiante 1 hora		F69B90F1	10:01	Estudiante 1 hora		F69B90F2	10:02	Estudiante 1 hora		F69B90F3	10:03	Estudiante 1 hora		F69B90F4	10:04	Estudiante 1 hora		F69B90F5	10:05	Estudiante 1 hora		 <p>19/11/2015 10:53:23 SERVICIO</p> <p>Boletos vendidos 105638 a 105663</p> <p>19/11/2015</p> <table border="1"> <tr> <td>Tarjeta 71E914C6</td> <td></td> </tr> <tr> <td>Corriente</td> <td>07:20</td> </tr> <tr> <td>1 hora</td> <td></td> </tr> <tr> <td>1 pasajeros</td> <td></td> </tr> <tr> <td>Tarjeta FD632D24</td> <td></td> </tr> <tr> <td>Estudiante</td> <td>07:21</td> </tr> <tr> <td>1 hora</td> <td></td> </tr> <tr> <td>1 pasajeros</td> <td></td> </tr> <tr> <td>Tarjeta 5B60AA91</td> <td></td> </tr> <tr> <td>Estudiante</td> <td>07:24</td> </tr> <tr> <td>1 hora</td> <td></td> </tr> <tr> <td>1 pasajeros</td> <td></td> </tr> <tr> <td>Tarjeta B92D5AD2</td> <td></td> </tr> <tr> <td>Estudiante</td> <td></td> </tr> </table> <p><i>Ilustración 12-131 Control de boleto después del cambio</i></p>	Tarjeta 71E914C6		Corriente	07:20	1 hora		1 pasajeros		Tarjeta FD632D24		Estudiante	07:21	1 hora		1 pasajeros		Tarjeta 5B60AA91		Estudiante	07:24	1 hora		1 pasajeros		Tarjeta B92D5AD2		Estudiante	
F69B90F0	10:00																																																				
Estudiante 1 hora																																																					
F69B90F1	10:01																																																				
Estudiante 1 hora																																																					
F69B90F2	10:02																																																				
Estudiante 1 hora																																																					
F69B90F3	10:03																																																				
Estudiante 1 hora																																																					
F69B90F4	10:04																																																				
Estudiante 1 hora																																																					
F69B90F5	10:05																																																				
Estudiante 1 hora																																																					
Tarjeta 71E914C6																																																					
Corriente	07:20																																																				
1 hora																																																					
1 pasajeros																																																					
Tarjeta FD632D24																																																					
Estudiante	07:21																																																				
1 hora																																																					
1 pasajeros																																																					
Tarjeta 5B60AA91																																																					
Estudiante	07:24																																																				
1 hora																																																					
1 pasajeros																																																					
Tarjeta B92D5AD2																																																					
Estudiante																																																					
<ul style="list-style-type: none"> ● Se rediseña los elementos de la lista siguiendo los consejos de destaque, alineación y jerarquía Padre/hijo/hermano para representar de mejor manera los datos. ● Se realizó un cambio de colores para que la vista vaya a tono con el color de la aplicación. ● Respetando que el usuario tenga que realizar la menor acción necesaria, cuando se controla una tarjeta de un pasajero, si pertenece al viaje, la lista realiza un <i>scroll</i> automático hasta la posición de la tarjeta controlada, resaltando el elemento para que el usuario lo encuentre más rápidamente. 																																																					

Tabla 12-71 Control de tarjeta de pasajeros de un viaje

Menú lateral

Antes

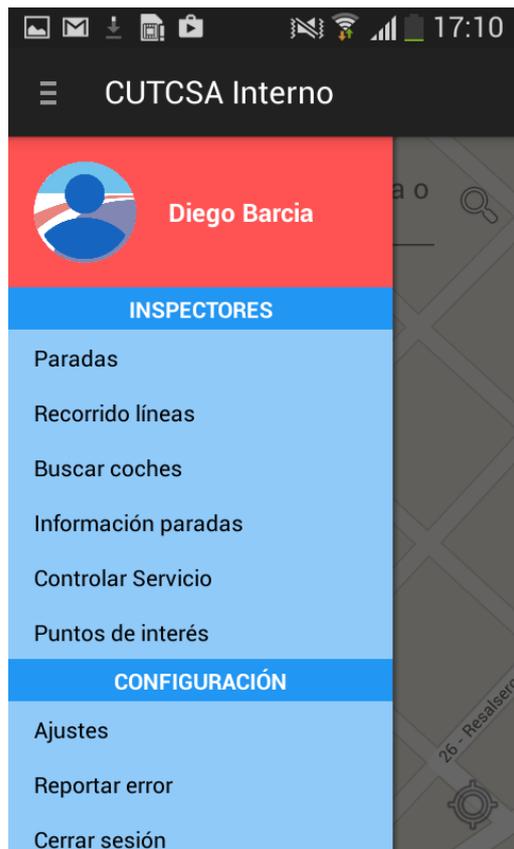


Ilustración 12-132 Menú lateral antes del cambio

Después

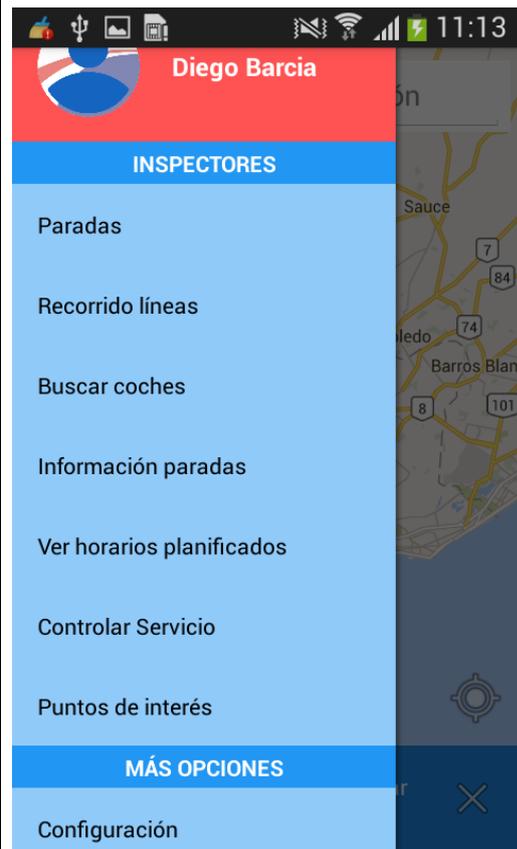


Ilustración 12-133 Menú lateral después del cambio

- Se cambia el título “configuración” por “más opciones” dado que es más representativo por su contenido (que albergaría ajustes, reportes, ayuda, cierre de sesión).
- Se aumenta el tamaño *clickable* de los elementos de la lista para que sea más preciso a la hora de seleccionar una opción.
- La opción “Ajustes” fue cambiada por “Configuración” siguiendo las sugerencias de diseño de Google en Material Design.
- Se agrega la funcionalidad de configuración, la cual no se encontraba en la versión que se le presentó a Daniel Mordeki en la reunión. Dicha opción permite al usuario activar y desactivar las notificaciones de sonido y vibración (las cuales fueron explicadas en la información del marcador de las Paradas):



Ilustración 12-134 Configuración después del cambio

Tabla 12-72 Menú lateral

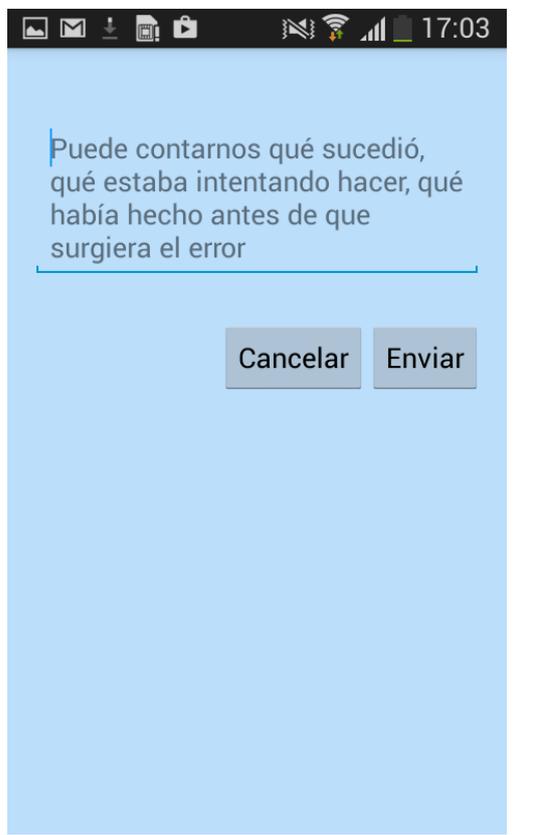
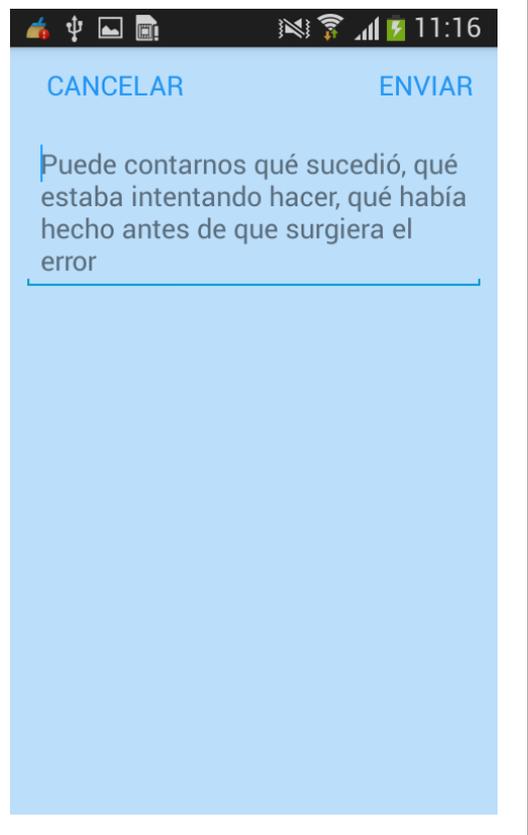
Envío de un reporte de error	
Antes	Después
 <p><i>Ilustración 12-135 Reporte de error antes del cambio</i></p>	 <p><i>Ilustración 12-136 Reporte de error después del cambio</i></p>
<ul style="list-style-type: none"> ● Se cambia la posición de los botones a donde usualmente se utiliza para este tipo de acciones. ● Se rediseña el tipo y diseño de los botones para que sea consistente con el resto de la aplicación. 	

Tabla 12-73 Envío de un reporte de error

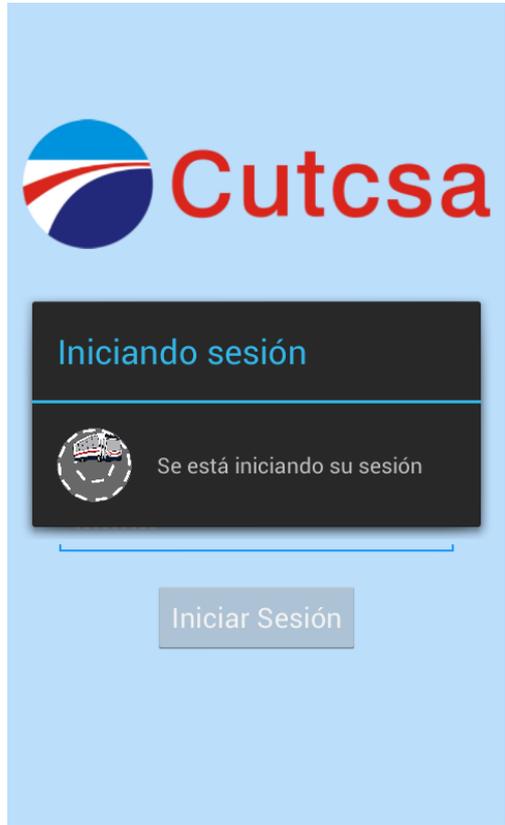
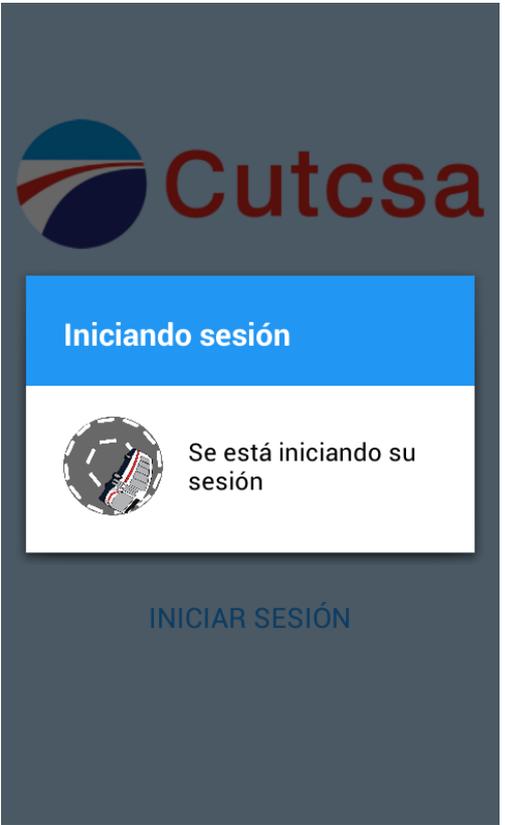
Mensaje de carga	
Antes	Después
 <p><i>Ilustración 12-137 Mensaje de carga antes del cambio</i></p>	 <p><i>Ilustración 12-138 Mensaje de carga después del cambio</i></p>
<ul style="list-style-type: none"> ● Se rediseña el mensaje de carga para que sea consistente con el resto de tipo de mensajes dentro de la aplicación. ● Se aumenta el tamaño de la imagen para que se entienda mejor qué es. 	

Tabla 12-74 Mensaje de carga

Mensaje correcto	
Antes	Después
 <p><i>Ilustración 12-141 Mensaje correcto antes del cambio</i></p>	 <p><i>Ilustración 12-142 Mensaje correcto después del cambio</i></p>
<p>No hubo modificaciones para este tipo de mensajes.</p>	

Tabla 12-76 Mensaje correcto

12.12. Anexo 12 - Análisis de Heurísticas de Nielsen

En este documento se responden las preguntas de la guía creada por Jakob Nielsen para asegurar la usabilidad de la aplicación.

- 1. Visibilidad del estatus del sistema: El sistema debe tener siempre informado a los usuarios de lo que está ocurriendo, con el *feedback* apropiado dentro del tiempo razonable.**

Las pantallas cuentan con títulos para que el usuario sepa en qué lugar de la aplicación se encuentra. En caso de ocurrir un error, el sistema reconoce la existencia del mismo y le informa al usuario de lo sucedido, dándole la oportunidad de reportar el mismo.

- 2. Correspondencia entre el sistema y el mundo real: El sistema debe hablar el lenguaje del usuario. Debe seguir convenciones del mundo real, hacer que la información aparezca en orden lógico y natural.**

En ningún momento la aplicación utiliza nomenclatura que no forma parte del lenguaje común, no existen mensajes ni etiquetas de error que dieran una excepción del sistema en lenguaje de programación, siempre se le presenta al usuario el cartel de error de la aplicación.

- 3. Control del usuario y libertad: los usuarios a menudo eligen funciones del sistema por error y necesitarán una “salida de emergencia” para dejar el estado no buscado sin necesitar ir a través de un lenguaje extendido. Se debe soportar *undo* y *redo*.**

Los teléfonos con sistema operativo Android tienen siempre un botón *undo* por defecto. Ese botón cumple con su funcionalidad en el programa de forma que si se desea volver a pantallas anteriores es posible. Por cuestiones de memoria no se guardan todas las pantallas las cuales el usuario visitó, sino que solamente las relacionadas a la funcionalidad que está realizando hasta el menú de inicio.

- 4. Consistencia y estándares: Los usuarios no deben preguntarse si diferentes acciones, palabras, situaciones o acciones significan la misma cosa. Seguir las convenciones de plataforma.**

De todas las acciones que aparecen en la aplicación ninguna parece ambigua ni repetitivas de otras acciones.

5. Prevención de errores: Mejor que mensajes de error es un cuidadoso diseño que prevenga la ocurrencia de problemas.

Para prevenir errores se realizó una interfaz fluida que permita al usuario fácilmente encontrar la funcionalidad deseada, esto se alcanzó gracias al menú lateral y controlando que en de 3 acciones pueda encontrar lo que desea. De esta forma es más difícil que el usuario se pierda en la aplicación y logré encontrar errores.

6. Reconocimiento sobre recuerdo: Hacer visible los objetos, las acciones y las opciones. El usuario no tendría que recordar información de una parte del diálogo a otra. Instrucciones para el uso del sistema debería ser visible o fácilmente recuperable cuando se lo requiera.

Las funcionalidades más complejas cuentan con un pequeño tutorial al comienzo en el cual se explica cómo se utiliza la funcionalidad. En caso de que quiera buscar información de ayuda, esta opción se encuentra en el menú lateral que está disponible en cualquier parte de la aplicación.

7. Flexibilidad y eficiencia de uso: Aceleradores -invisibles para los usuarios novatos- pueden acelerar la interacción del usuario experto de forma que el sistema, sirva tanto al novato como al experto.

Al usuario se le presenta siempre la mayor cantidad de información posible sin abrumar al mismo. Esto simplifica el trabajo de buscar información y navegar innecesariamente por la aplicación. Por ejemplo, cuando se ve una parada en el mapa, ya indica cual es el próximo arribo de la misma, otro caso es cuando hay nuevas notificaciones se le indican con un sonido y queda en el menú lateral que tiene notificaciones sin leer.

8. Estética y diseño minimalista: Los diálogos no deben contener información irrelevante o raramente necesaria. Cada unidad extra de información compete con la información relevante y disminuye su visibilidad relativa.

La información que se muestra en las pantallas fue validada por el cliente. Por lo que toda información es relevante para el usuario.

9. Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de los errores: Los mensajes de error deberían ser expresados en lenguaje llano, indicar de forma precisa el problema y sugerir una solución de forma constructiva.

Existen mensajes de error para casos particulares en el cual se le presenta al usuario cual fue el error sucedido.

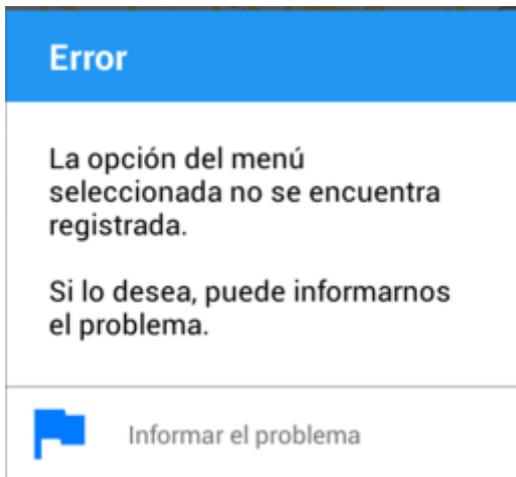


Ilustración 12-143 Mensaje de error en la aplicación

En caso de surgir un error del sistema se le informa al usuario. Este siempre que recibe un mensaje de error tiene la opción de reportar el problema. Esto va a ayudar a mejorar la aplicación y reducir errores.

- 10. Ayuda y documentación: Aun pensando que es mejor si el sistema puede ser usado sin documentación, puede ser necesario proveer ayuda y Documentación. Cualquier información de este tipo debería ser fácil de buscar, debería estar focalizada en las tareas del usuario, listar pasos concretos, y no ser demasiado larga.**

La aplicación cuenta con una sección de ayuda ubicada en el menú lateral. En esta se encuentran las guías dinámicas de ayuda para las funcionalidades complejas. En caso de necesitar más guías, la aplicación está diseñada para poder agregarlas fácilmente.

12.13. Anexo 13 - Análisis de problemas de visión cromática

12.13.1. Situación

Existen muchos problemas que pueden interferir en la forma en que se muestra la información dentro de la aplicación. Uno de ellos es el problema de visión, que incluye el Daltonismo. Dicho problema refiere a la incapacidad de un individuo de diferenciar entre algunos colores, y afectan principalmente a los hombres (1 de cada 10 tienen algún problema de daltonismo). Existen varios grados de nivel de problemas de visión cromática, que incluye:

- dificultad para diferenciar entre el amarillo y el verde (Tritanopía).
- dificultad para diferenciar entre el azul y el verde (Tritanopía o Protanopía).
- dificultad para diferenciar entre el rojo y el verde (Protanopía).
- dificultad para diferenciar entre el rojo y el púrpura (Deuteranopía).
- dificultad para diferenciar entre el verde y el púrpura (Deuteranopía).
- ve todos los colores en tonos de grises (Acromatopsia) [46].

12.13.2. Solución

Para poder ayudar a diferenciar distinta información o estados de la aplicación, se resolvió jugar con los colores Azul, Rojo, Blanco y Negro, dado que en todos los casos (a excepción de la Acromatopsia) no habría problema de distinguir entre dichos colores. Este análisis se puede ver en mayor profundidad en el Anexo.

Aun así, siempre que sea posible, se incluirá información escrita (utilizando alto contraste entre el fondo y el texto, tal como se describe en la sección de “opacidad para texto, íconos y divisores” [47]).

12.13.3. Bibliografía

[1] Medline Plus, «Medline Plus Información de salud de la biblioteca nacional de medicina,» 2015. [En línea]. Available: <https://www.nlm.nih.gov/medlineplus/spanish/ency/article/003387.htm>. [Último acceso: 11 2015].

[2] Google, «Google Material design,» 2015. [En línea]. Available: www.google.com/design/spec/style/color.html#color-ui-color-application. [Último acceso: 11 2015].

[3] Wikipedia, «Wikipedia,» 2015. [En línea]. Available: <https://es.wikipedia.org/wiki/Daltonismo>. [Último acceso: 11 2015].

12.13.4. Anexo

Para cada una de las patologías, se realiza la comparación entre los colores cómo los vería una persona con visión normal, y cómo los vería una persona con el problema. En cada caso, se destacan los colores Rojo y Azul (de la visión normal) y su correspondiente para la visión afectada, y así realizar el análisis en base a dichos colores.

Visión con Protanopía	
Visión normal	Visión afectada
	
	
	
	
	
	
<i>Ilustración 12-144 Visión Normal Protanopía</i>	<i>Ilustración 12-145 Visión afectada Protanopía</i>
Si tomamos en cuenta cómo el individuo ve el color Rojo y el Azul, comparado con la visión normal, vemos que ambos colores los podría diferenciar sin problemas.	

Tabla 12-77 Visión con Protanopía

Visión con Deuteranopía	
Visión normal	Visión afectada
 <p><i>Ilustración 12-146 Visión Normal Deuteranopía</i></p>	 <p><i>Ilustración 12-147 Visión afectada Deuteranopía</i></p>
<p>Si tomamos en cuenta cómo el individuo ve el color Rojo y el Azul, comparado con la visión normal, vemos que ambos colores los podría diferenciar sin problemas.</p>	

Tabla 12-78 Visión con Deuteranopía

Visión con Tritanopía	
Visión normal	Visión afectada
 <p><i>Ilustración 12-148 Visión Normal Tritanopía</i></p>	 <p><i>Ilustración 12-149 Visión afectada Tritanopía</i></p>
<p>Si tomamos en cuenta cómo el individuo ve el color Rojo y el Azul, comparado con la visión normal, vemos que ambos colores los podría diferenciar sin problemas.</p>	

Tabla 12-79 Visión con Tritanopía

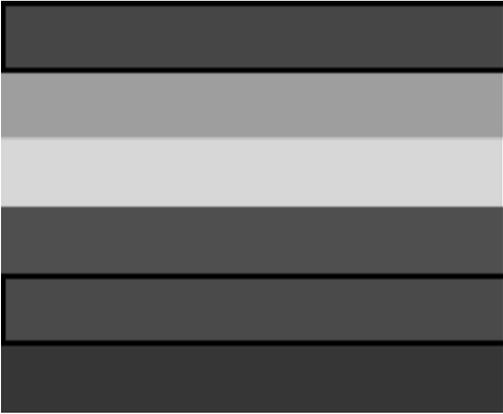
Visión con Acromatopsia	
Visión normal	Visión afectada
	
<i>Ilustración 12-150 Visión normal Acromatopsia</i>	<i>Ilustración 12-151 Visión Afectada Acromatopsia</i>
<p>Si tomamos en cuenta cómo el individuo ve el color Rojo y el Azul, comparado con la visión normal, vemos que ambos colores son muy difíciles de distinguir, ya que son dos tonos muy similares de grises.</p>	

Tabla 12-80 Visión con Acromatopsia

Imágenes obtenidas de [48]

12.14. Anexo 14 - Estudio Facilidad Aprendizaje

El objetivo del siguiente estudio es verificar si con un determinado tiempo de entrenamiento los usuarios del sistema son capaces de aprender ciertas funcionalidades de forma satisfactoria. El estudio se realizó sobre una muestra de empleados quienes serán futuros usuarios de la aplicación.

Empleados utilizados en el estudio: 7.

Fecha de realización: jueves 18 de noviembre de 2015.

Se realizó el estudio con 2 de las principales funcionalidades del sistema:

- C2-RF7: Control de Servicio
- C2-RF3: Mostrar arribo de líneas a la parada

Se comenzó por explicarle a los empleados como usar las funcionalidades a probar utilizando una pequeña demo que fue proyectada, y donde se dedicó 10 minutos a explicar cada una de las funcionalidades y además se destinó 5 minutos para atender preguntas por parte de los usuarios.

Control de Servicio

Objetivo de la prueba	Tiempo estimado para completar la prueba
Obtener los datos necesarios para realizar un control de servicio desde la expendedora de billetes utilizando una tarjeta NFC puente y lograr controlar 1 tarjeta STM con viaje valido y otra con un viaje invalido.	4 minutos.

Tabla 12-81 Objetivos y tiempo estimado

Nro. Prueba	Tiempo real (minutos)	¿Pudo terminar la prueba?	Observaciones
1	2	SI	
2	2	SI	
3	3	SI	
4	4	SI	
5	3	SI	
6	5	SI	Problemas con la lectura de la tarjeta puente.
7	2	SI	

Tabla 12-82 Ejecución de pruebas Control de servicio

Mostrar arribo de líneas a parada

Objetivo de la prueba	Tiempo estimado para completar la prueba
Seleccionar una parada determinada, luego seleccionar una línea determinada y poder visualizar cuando va a ser el próximo arribo de la misma.	2 minutos.

Tabla 12-83 Objetivo y tiempo estimado

Nro. Prueba	Tiempo real (minutos)	¿Pudo terminar la prueba?	Observaciones
1	2	SI	
2	2	SI	
3	1	SI	
4	3	SI	Persona no muy hábil con teléfonos móviles.
5	1	SI	

6	2	SI	
7	1	SI	

Tabla 12-84 Ejecución de pruebas Mostrar arribo de líneas a parada

De este estudio se puede concluir que, con un entrenamiento de 25 minutos para 2 funcionalidades del sistema, se logró que la totalidad de los usuarios aprendieran a utilizar dichas funcionalidades de forma correcta. Si bien hubo 2 casos en los que llevo más tiempo del previsto, de todos modos, estos lograron culminar la prueba.

12.15. Anexo 15 - Prueba de carga y eficiencia

El objetivo de la prueba fue probar el comportamiento del sistema ante una situación de stress y verificar si inclusive en esta situación el mismo sigue siendo respetando los parámetros de eficiencia definidos.

La herramienta utilizada para realizar las pruebas fue el módulo *LoadTest* del Microsoft Visual Studio Ultimate 2012.

Como fue definido en el capítulo de Arquitectura, el sistema debería de soportar a 16 usuarios concurrentemente. Por lo cual se diseñó una prueba de carga y eficiencia con las siguientes características:

- 16 usuarios concurrentes.
- Peticiones a 10 servicios diferentes del sistema al azar.
- 10 minutos de duración.

En la siguiente Ilustración se pueden ver los servicios cargados en la prueba.

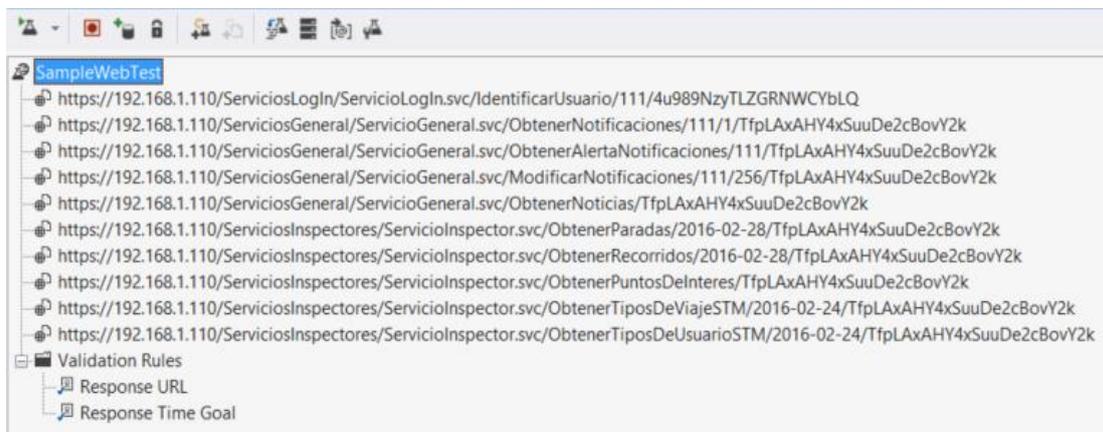


Ilustración 12-152

Ya que la empresa decidió no exponer los servicios en internet por el momento, lo cual limitó las posibilidades de poder probar el sistema en un entorno real de uso. Para poder tener una prueba más cercana a la realidad, se configuró que se simule la conexión como si fuese mediante 3G, tal como sería si se comunicara un móvil en la calle con el *backend*. En la siguiente Ilustración se puede ver dicha configuración.

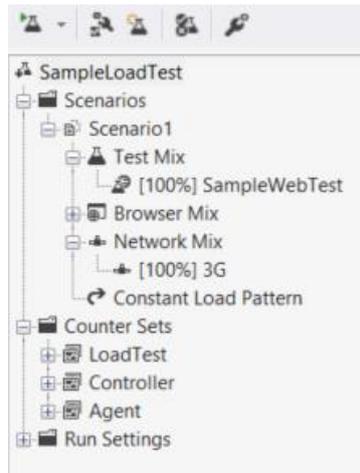


Ilustración 12-153 Configuración 3G

Luego de correr el test, se pudo establecer que el tiempo de respuesta promedio de los servicios fue de 0.68 segundos y el tiempo máximo fue de 0.89 segundos. Valores inferiores a los 5 segundos definidos como restricción del sistema.

En el siguiente Gráfico se puede ver a modo de resumen los tiempos de respuesta de los distintos servicios a lo largo de la prueba de 10 minutos de duración y de un total de 14.040 peticiones, todas fueron menores a 1 segundo. El eje X representa el tiempo transcurrido y el eje Y representa el tiempo de respuesta a las peticiones.

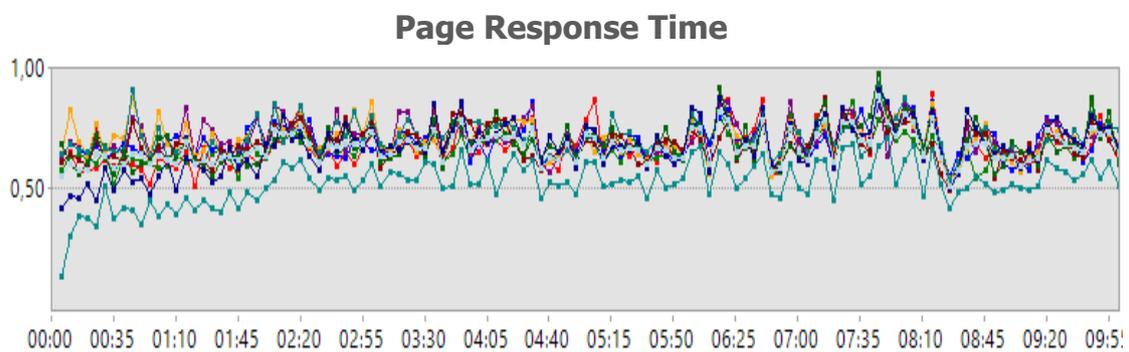


Gráfico 12-19 Tiempos de respuesta de los servicios

El resumen de la ejecución de las pruebas se puede visualizar en la siguiente Ilustración, de donde se extrae que no hubo ninguna prueba fallida. Por lo cual se concluye que el sistema pudo soportar sin problemas los 16 usuarios concurrentes cumpliendo a la vez con la restricción de 5 segundos máximos de tiempo de respuesta.

Overall Results

Max User Load	16
Tests/Sec	2,34
Tests Failed	0
Avg. Test Time (sec)	6,79
Transactions/Sec	0
Avg. Transaction Time (sec)	0
Pages/Sec	23,6
Avg. Page Time (sec)	0,68
Requests/Sec	23,6
Requests Failed	0
Requests Cached Percentage	0
Avg. Response Time (sec)	0,68
Avg. Content Length (bytes)	112

Ilustración 12-154 Resumen de ejecución de prueba

12.16. Anexo 16 - Plan SCM

12.16.1. Objetivos

El plan de SCM define como se hará el control de la configuración de los elementos de *software* que componen a los productos a lo largo del ciclo de vida del proyecto. Para ello se identifican los diferentes elementos de configuración, se define como se van a controlar y versionar, y finalmente se define el proceso de gestión de control de cambios de los mismos.

12.16.2. Definiciones

Repositorio: Es el espacio físico en donde se almacenan todos los elementos de configuración generados durante el proceso de desarrollo en sus diferentes versiones. Estos elementos pueden estar en tres estados: Pendientes de aprobación, Aprobados y en Producción.

SCM: Se denomina Gestión de la Configuración de software al conjunto de procesos destinados a asegurar la calidad de todo producto obtenido durante cualquiera de las etapas del desarrollo de un sistema a través del estricto control de los cambios realizados sobre los mismos y de la disponibilidad constante de una versión estable de cada elemento para toda persona involucrada en el dicho desarrollo

SCI: Elemento de Configuración del Software. Son los elementos creados durante el proceso Pueden ser de tres tipos: 1) De Software: código fuente, recursos gráficos, bibliotecas, ejecutables. 2) Documentos: técnicos, administrativos y de usuario. 3) Estructuras de datos: estructura de base de datos, datos iniciales, archivos de configuración.

12.16.3. Responsabilidades

Se elaboró la siguiente tabla para poder detallar las responsabilidades de cada integrante del equipo en lo que refiere a la gestión de configuración del proyecto.

Rol	Nombre	Responsabilidad
Líder de SCM	Andrés Amegeiras	<ul style="list-style-type: none">- Respaldos.- Instruir al resto del equipo en usar el repositorio de forma adecuada.

		<ul style="list-style-type: none"> - Instalación y configuración de las distintas herramientas tecnológicas que se requieren para el repositorio y su uso.
Demás integrantes del equipo	Diego Barcia y Juan Pedro Trecca	<ul style="list-style-type: none"> - Utilizar durante la ejecución del proyecto las herramientas de forma correcta. - Utilizar el formato establecido al momento de liberar un reléase nuevo.

Tabla 12-85 Roles y responsabilidades

12.16.4. Identificación de los elementos de la configuración

Se identificaron 4 tipos distintos de elementos de configuración del software (ECS), los cuales se desprenden del proceso de Ingeniería y de los procesos de apoyo, los mismos se listan a continuación:

- Documentos de trabajo: Son documentos que sirven para el trabajo de los integrantes pero que no serán entregados para su evaluación.
- Documentos formales: Son documentos formales que necesitan más formalidad y revisiones ya que serán entregados para su evaluación.
- Multimedia: Son fotos, videos o audio registrados durante el transcurso del proyecto que es necesario tener disponibles para futura referencia.
- Código Fuente: Son todos los elementos que comprenden al código fuente, sea del *frontend* o *backend* del sistema.

12.16.5. Selección de herramientas

Debido a la naturaleza del proyecto, la elección de las herramientas se hizo en base a las categorías de archivos que se describieron en el apartado anterior

12.16.5.1. Documentos de trabajo, Documentos formales y multimedia

Para los elementos de configuración relativos a los Documentos de trabajo se buscaron herramientas que cuenten con las siguientes características:

- Historial de versiones por documento.
- Posibilidad de revertir a una versión anterior de un documento.
- Sincronización con la nube con el fin de acceder desde cualquier lugar y tener un respaldo robusto.
- Compatibilidad con formato de archivos de Microsoft Office.

Para los ECS relativos a los Documentos formales se buscaron herramientas que cuenten con las siguientes características:

- Historial de versiones por documento.
- Posibilidad de revertir a una versión anterior de un documento.
- Sincronización con la nube con el fin de acceder desde cualquier lugar y tener un respaldo robusto.
- Compatibilidad con formato de archivos de Microsoft Office.
- Edición simultánea online del documento por parte de los integrantes del equipo.

Para los ECS relativos a los archivos Multimedia se buscaron herramientas que cuenten con las siguientes características:

- Historial de versiones por archivo.
- Sincronización con la nube con el fin de acceder desde cualquier lugar y tener un respaldo robusto.

Herramientas analizadas:

Google Drive: Ofrece 5GB de almacenamiento gratuitos, además de poder editar documentos de Microsoft Office online. Cuenta con reproductores online para varios formatos de archivos multimedia. Es una herramienta muy utilizada por los integrantes del equipo, por ende, están muy familiarizados con la misma.

Microsoft OneDrive: Ofrece 10GB de almacenamiento gratuitos, y posee la mejor integración con documentos Microsoft Office comparado con otros servicios de almacenamiento en la nube. Ya que permite editar simultáneamente con otros usuarios

un mismo documento, pero utilizando directamente las aplicaciones office de escritorio, por ende, siempre se va a mantener el formato de los documentos.

Se seleccionó Google drive para los Documentos de trabajo y multimedia, ya que este tipo de archivos son los que más se van a utilizar a lo largo de proyecto y dado que no se necesita tener tanta formalidad con los formatos, y es la herramienta más usada por todos, es la mejor opción.

Para los Documentos formales se seleccionó Microsoft OneDrive debido a su integración con Microsoft Office de escritorio, la cual es superior a la de Google Drive.

12.16.5.2. Código fuente

Los ECS relativos a los archivos Código Fuente se buscaron herramientas que cuenten con las siguientes características:

- El código sea privado para los integrantes del equipo de proyecto.
- Acceso para al menos 3 desarrolladores.
- Preferentemente gratuito.
- Posibilidad de recuperar versiones anteriores.
- Integración con Microsoft Visual Studio para el *backend*.
- Integración con Android Studio para el *frontend*.

Herramientas analizadas:

Bitbucket: Permite alojar código en la nube, además de contar con herramientas de escritorio y consola. Es gratuito y como principal ventaja el código es privado.

GitHub: Muy similar a Bitbucket, pero para poder tener el código de forma privada, hay que pagar.

Microsoft TFS: Repositorio gratuito con integración nativa con Microsoft Visual Studio (es decir, no se necesita un versionador externo). El código es privado y es gratuito hasta 5 usuarios.

Sourcetree: Versionador de código gratuito el cual se puede integrar con Bitbucket.

En un principio se valoró más el intentar utilizar un solo repositorio tanto para alojar el *backend* como el *frontend*, por lo cual se había elegido para el código del *frontend*: Bitbucket, el cual tiene integración nativa con Android studio y además el código es

privado de forma gratuita. Y para el *backend* Bitbucket junto con Sourcetree. La integración con Microsoft Visual Studio no resulto muy amigable y se tuvo varios problemas al momento de integrar soluciones. Por lo cual se decidió cambiar el repositorio del *backend* por Microsoft TFS.

La siguiente tabla se resume las herramientas seleccionadas por cada elemento de configuración.

Elemento de configuración	Herramienta
Documentos de trabajo	Google Drive
Documentos formales	Microsoft OneDrive
Multimedia	Google Drive
Código Fuente	Bitbucket para el <i>frontend</i> Microsoft TFS para el <i>backend</i>

Tabla 12-86 Herramientas seleccionadas

12.16.6. Gestión del repositorio

Para los repositorios de la documentación se tuvieron 2 grandes consideraciones, por un lado, se separó en carpetas y se versionó los documentos que debieron de ser mostrados y enviados para su aprobación, separando las versiones enviadas a aprobar de las ya aprobadas.

Por otro lado, debido al ciclo de vida evolutivo elegido, algunos documentos, principalmente el ESRE se vieron modificados en cada evolución. Por lo cual se crearon carpetas separando archivos específicos de cada una de estas. En la siguiente ilustración se puede ver la estructura de directorios utilizada.

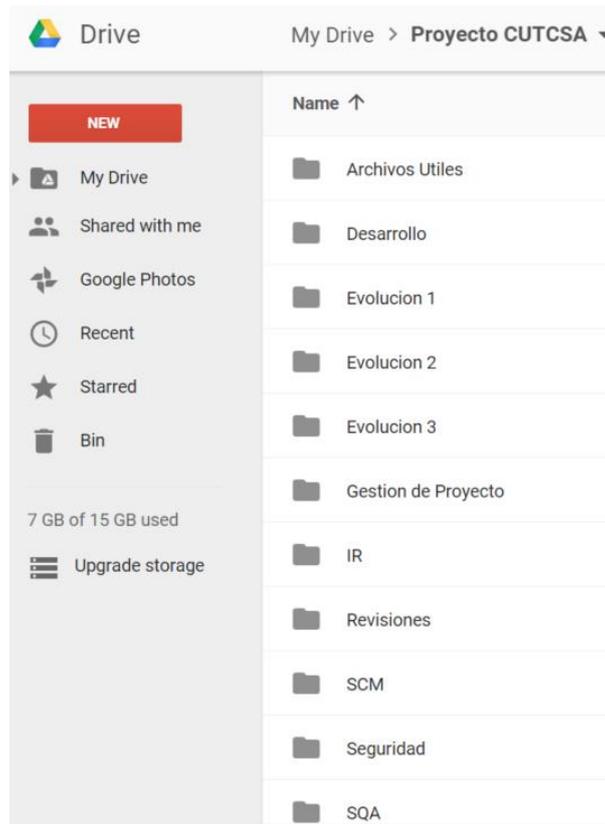


Ilustración 12-155 Estructura repositorio documentos, multimedia

Para los repositorios del código fuente, debido a que es un equipo pequeño, y a que en general se desarrolló de forma bastante desacoplada, se decidió no utilizar ramificaciones del código, y además también utilizar solamente dos repositorios, uno para la aplicación Android y otro para el *backend*, es decir, no utilizar múltiples repositorios por cada sub componente del sistema.

En la siguiente Ilustración se puede visualizar el repositorio en Bitbucket de la aplicación móvil, en donde se pueden ver algunos *commits* realizados por integrantes del equipo.

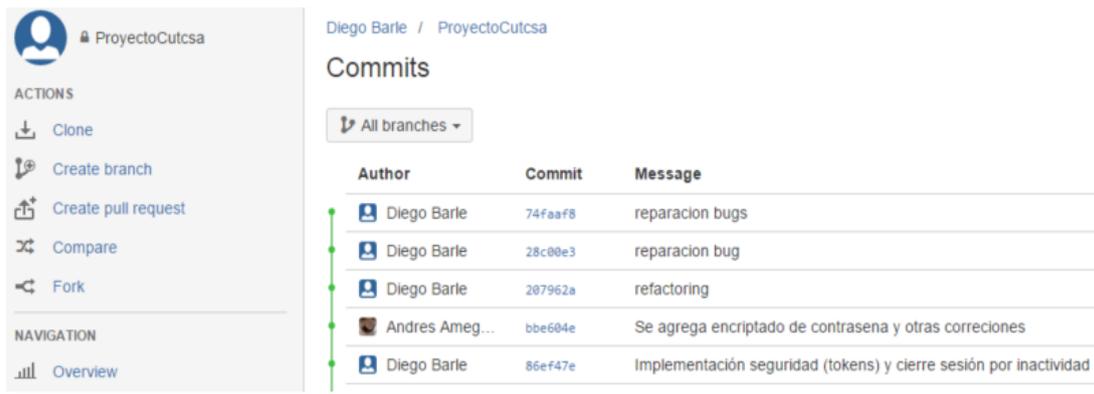


Ilustración 12-156 Repositorio frontend

12.16.7. Release de versiones

A lo largo del desarrollo, en los momentos en los que se alcanzaron ciertos hitos, o se desarrollaron funcionalidades específicas, se liberaron versiones para que se pueda probar con ellas o usarlas para validar con el cliente.

El estándar utilizado para nombrar los componentes liberados del *backend* fue: BXX.YY, donde XX representa un número de versión mayor, el cual cambia cuando se agregan grandes nuevas funcionalidades, YY representa un número de versión menor, el cual cambia a medida que se agregan o arreglan funcionalidades menores.

Análogamente para los componentes liberados del *frontend*, se utiliza el siguiente estándar: FXX.YY.

Dado que el sistema está compuesto por ambos componentes, se definió una tabla en la cual se especificaban las versiones de *backend* y *frontend* que juntas empaquetadas formaban una versión de reléase, para la cual se utilizó el siguiente estándar: VXX, siendo XX un número. En la siguiente Tabla se puede observar un ejemplo:

Release	Versión backend	Versión frontend
V01	B02.05	F02.06
V02	B02.08	F03.01

Tabla 12-87 Release de versiones

12.16.8. Respaldos y actualizaciones

Se decidió que 2 veces al mes, la primera en los días cercanos al principio de cada mes y la segunda sobre mediados, se va a copiar todo el contenido de OneDrive, Google Drive, Bitbucket y TFS. y será almacenado de forma cifrada en un servidor privado.

Estos respaldos se van a rotular de la siguiente manera “X - Backup – FechaDelBackup” en donde el “X” representa un identificador, el cual va a ser un número creciente comenzando en 1. Esta tarea va a ser realizada por Andrés Amegeiras.

Estos respaldos no serán utilizados a menos que no alcance con intentar obtener una versión anterior con las herramientas de versionado utilizadas, de ser así, un usuario podrá solicitar al líder de SCM una versión anterior.

12.17. Anexo 17 - Pruebas de seguridad

El objetivo de estas pruebas es el testeo del correcto funcionamiento de todo lo implementado relativo a seguridad del sistema.

La prueba fue realizada como validación por parte de la empresa de que la aplicación cumple con los requisitos de seguridad esperados.

La prueba fue ejecutada en conjunto con el administrador de redes de la empresa, quien además se encarga de todos los aspectos relativos a la seguridad de los datos y comunicaciones.

La prueba se dividió en 3 grandes partes:

- Chequear el correcto funcionamiento del manejo de la sesión de usuario.
- Chequear que el encriptado de la contraseña de usuario.
- Chequear que las comunicaciones se están llevando a cabo mediante un canal seguro.

Fecha de realizada: miércoles 13 de enero de 2016.

Sesión de usuario

Se plantearon algunas pruebas relativas a la sesión de usuario, donde se probó que la aplicación no dejara operar a un usuario en caso de su sesión haber expirado ya sea por *timeout* o inactividad.

Cabe aclarar que los valores de cierre de sesión por *timeout* o inactividad son configurables, habiéndose configurado por sugerencia del administrador de redes de la empresa, 8 horas para el *timeout* y 30 minutos para la inactividad.

¿Qué se prueba?	Resultado esperado	Resultado obtenido
Que la sesión de un usuario el cual se logueo hace más de ocho horas se cierre por al intentar	Se cierra la sesión del usuario, se muestra la ventana de Log in y se le	El esperado.

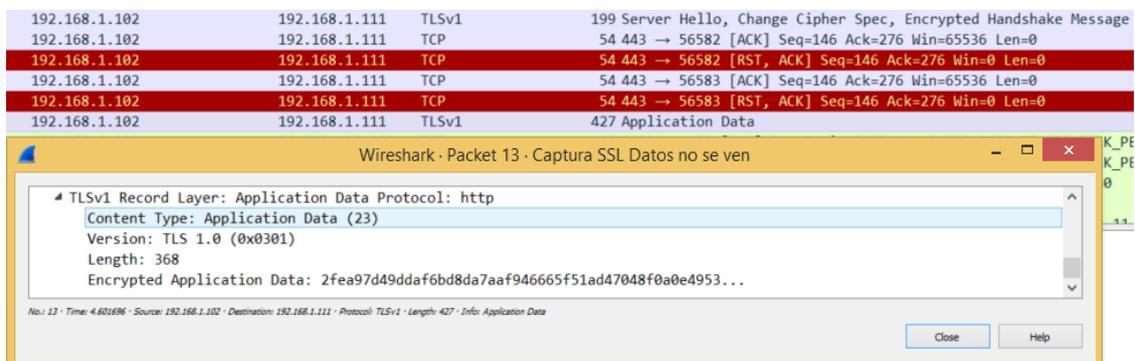
operar en el sistema nuevamente.	informa al usuario de lo ocurrido.	
Que la sesión de un usuario el cual se logueo y no opero en el sistema por 31 minutos se cierre al intentar operar nuevamente.	Se cierra la sesión del usuario, se muestra la ventana de Log in y se le informa al usuario de lo ocurrido.	El esperado.

Tabla 12-88 Pruebas sesión

Canal seguro (TLS/SSL)

Utilizando la herramienta de análisis de tráfico de red, Wireshark, se capturo el paquete donde viajan los datos enviados desde la aplicación móvil hacia el *backend*, luego de establecida la conexión, para intentar ver el contenido del mismo y corroborar la correcta implementación del canal seguro.

En la siguiente ilustración se muestra el paquete capturado, donde se puede ver que, en la sección correspondiente a los datos, los mismos se encuentran encriptados (“Encrypted Application Data: 2fea97d..”), además al observar la configuración del paquete, se puede ver que utiliza TLS= *Transport Layer Security*.



Encriptado de contraseña

Se corrobora que en la base de datos la contraseña de los usuarios se encuentra encriptada, en este caso se utilizó como algoritmo de encriptación AES con una clave de 128 bits de tamaño, la cual es configurable. En la siguiente ilustración se puede observar el contenido de la tabla usuario de la base de datos, donde se confirma que la contraseña esta encriptada.

Usuarioid	Nombre	Apellido	Contraseña	NumeroF...	Rolld	CargoFunci...
2	Pepe	Lopez	4u989NzyT/LZGRNWC+YbLQ==	123	1	1

Ilustración 12-157 Tabla "Usuario" en la Base de datos de la aplicación

Además de esto, se configuro el Log in de la aplicación para que no use un canal seguro momentáneamente (http en vez de https), y así poder capturar el paquete de datos enviado desde la aplicación móvil y corroborar que la contraseña también viaja encriptada.

Efectivamente, la contraseña que se vio en los datos del paquete es la misma que la mostrada en la última Ilustración. Por lo que viaja encriptada.

Los 3 tipos de prueba realizadas fueron exitosos, por lo que se concluye que la implementación de seguridad del sistema es correcta.

12.18. Anexo 18 - Aplicación Manage It

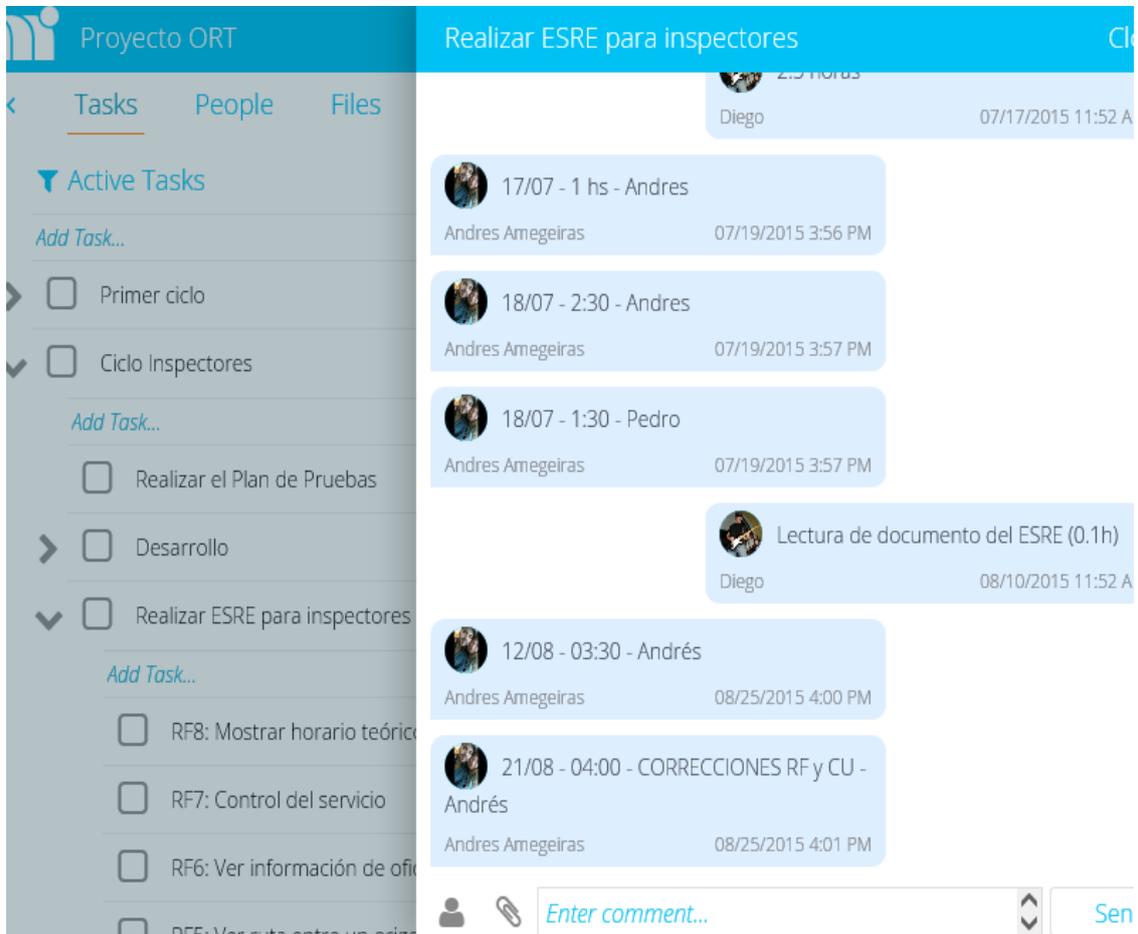


Ilustración 12-158 Imagen de la aplicación

12.19. Anexo 19 - Registro de horas

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	FECHA	HORAS	SUMA SEMANAL	HORAS G. PROY.	HORAS ING. REQ.	HORAS INV. Y ENT.	HORAS ARQ.	HORAS DES.	HORAS TEST	HORAS SQA	HORAS SCM	HORAS REV. Y DOC. PROY.	HORAS RETRABAJO	DESCRIPCIÓN TAREA 1
53	miércoles, 27 de mayo de 2015	3,5		3,5										Armado planilla de Gestión de Entregables (0,5h)
54	jueves, 28 de mayo de 2015	5		2,5	1,5		1							Corregir diagramas de proceso (1.25h)
55	viernes, 29 de mayo de 2015	3,6		1,5	1,1	1								Envío mail a entrevistados que falta recibir feedback (0.2h)
56	sábado, 30 de mayo de 2015	0												
57	domingo, 31 de mayo de 2015	0												
58	lunes, 1 de junio de 2015	3,5	21,25	3	0,5									Corrección a entrevista con jefe de transporte y Borrador del Plan (0.5h)
59	martes, 2 de junio de 2015	3,5		3,5										Continuación Plan del Proyecto (3.5h)
60	miércoles, 3 de junio de 2015	6,25		2	2,75		1,5							Continuación Plan del Proyecto (1h)
61	jueves, 4 de junio de 2015	3,5		3	0,5									Pasar la reunión con tutor (0.5h)
62	viernes, 5 de junio de 2015	3,5		3,5										Correcciones Plan del Proyecto (3.5h)
63	sábado, 6 de junio de 2015	0,5		0,5										Reunión con Pedro para explicarle planes (0.5h)
64	domingo, 7 de junio de 2015	0,5			0,5									Mostrarle "Borrador del Plan" al Gte. General (0.5h)
65	lunes, 8 de junio de 2015	3,95	29,3	2			1,5			0,25		0,2		Correcciones Plan del Proyecto (2h)
66	martes, 9 de junio de 2015	3,75			0,25	0,75	1,25			1,5				Refactoring y subida al drive de la prueba Place Autocomplete de Google (0.75h)

Ilustración 12-159 Registro de horas de uno de los integrantes usando planillas de Google

12.20. Anexo 20 - ToDoList

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Tarea		Subtareas Pendientes	Prioridad	Estado	Referencia	Finalización esperada	Finalización real	Responsable	Horas Diego	Horas Pedro	Horas Andrés	Actividad del Proyecto
2	Ciclo 1				EJEMPLO		28/06/2015						
3		Tarea 1		Muy Alta	EJEMPLO	ESRE	25/06/2015		Diego	15	2	0	DES
4		Tarea 2		Alta	EJEMPLO	ESRE	26/06/2015		Andres	0	0	17	ING REQ
5		Tarea 3		Media	EJEMPLO	ESRE	28/06/2015		Pedro	3	16	1	DES
6	Ciclo 2				EJEMPLO		10/07/2015						
7		Tarea 1		Baja	EJEMPLO	ESRE	10/07/2015		Diego	0	0		GESTION
8		Identificar entregables del producto y del proyecto		Alta	Iniciado	Reunión 2 con tutor	Semana 8		Pedro				SQA
9		Definir Plan del proyecto		Muy Alta	Iniciado	Reunión 3 con tutor	Semana 8		Diego				GESTION
10		Definir Plan de SCM		Muy Alta	Iniciado	Reunión 4 con tutor	Semana 7		Andrés				SCM
11		Definir plan de SQA (primera versión)		Muy Alta	Iniciado	Reunión 4 con tutor	Semana 7		Pedro				SQA
12		Realizar presentación y ppt de la primera Revisión del Proyecto		Muy Alta	Iniciado	Reunión 4 con tutor	Semana 8		Todos				REV. Y DOC. PROY.
13		Definir reunión con programadores y analistas para ver arquitectura y realizar relevamiento de RNF		Media	No Iniciado	Reunión 4 con tutor	Semana 10		Diego				GESTION
14		Averiguar sobre alguna empresa que nos pueda dar una mano con UI		Baja	Iniciado	Reunión 4 con tutor	Semana 10		Diego				ARQUITECTURA
15		Definir reunión con Sub Gerente General		Baja	Finalizado	Reunión 4 con tutor	Semana 5	Semana 8	Diego				GESTION
16		Definir el plan de calidad		Muy Alta	Finalizado	Reunión 4 con tutor	Semana 7	Semana 7	Pedro				SQA
17		Crear .apk de la aplicación de Inspectores		Baja	Finalizado	-	Semana 8	Semana 8	Diego				ING. REQ.
18		Definir roles y responsables		Muy Alta	Finalizado	Reunión 1 con tutor	Semana 2	Semana 1	Todos				GESTION
19		Obtener información sobre la empresa, áreas		Muy Alta	Finalizado	Reunión 1 con tutor	Semana 2	Semana 2	Andrés				ING. REQ.
20		Definir suplentes de roles		Baja	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Todos				GESTION
21		Preguntar por el área de CAC		Baja	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Diego				ING. REQ.
22		Preguntar por metodología de trabajo en CUTCSA		Media	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Diego				GESTION
23		Averiguar sobre la metodología MSF		Media	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Pedro				SQA
24		Definir actores a quiés es clave hacer entrevistas para obtener una visión inicial del área		Muy Alta	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Andrés				ING. REQ.
25		Definir entrevistas con actores, junto con preguntas que vamos a hacerles		Muy Alta	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Andrés				ING. REQ.
26		Definir días de reunión entre nosotros		Alta	Finalizado	Reunión 2 con tutor	Semana 3	Semana 2	Todos				GESTION
--		Discutir entre usar MSF o ESRE básico.											

Ilustración 12-160 Herramienta utilizada al inicio del proyecto para gestión de tareas

12.21. Anexo 21 – Reunión final con el Sub Gerente General de C.U.T.C.S.A.

El día 23 de febrero del 2016, se tuvo una reunión con el Sub Gerente General de C.U.T.C.S.A., Álvaro Santiago, con el objetivo de describir de forma resumida los objetivos del proyecto, y el desarrollo realizado en su totalidad.

Luego de la reunión, se le formularon un conjunto de preguntas para obtener su opinión general del proyecto. Las respuestas se detallan a continuación.

- **A grandes rasgos, ¿qué opinión merece el proyecto que realizamos durante este año?**

“En primer lugar, considero que se ofrece una herramienta que permitirá mejorar la gestión de áreas muy sensibles de la empresa desde el punto de vista de la gestión de la flota y de la información a nuestros clientes (externos e internos).

En segundo lugar, esa mejora de gestión resultará de contar con información más completa y en tiempos muy útiles para simplificar las tareas del personal, por lo que entiendo que es muy positivo desde el punto de vista del aumento de la eficiencia en el cumplimiento de las tareas habituales.”

- **¿Le parece que la aplicación desarrollada se alinea con la misión, visión y valores de C.U.T.C.S.A.?**

“Desde luego, contar con información oportuna y confiable para una mejor toma de decisiones nos permite ofrecer una mejor calidad de servicio a nuestros clientes y, naturalmente, el retorno asociado a estas mejoras en términos de condiciones de trabajo para nuestro personal y en términos de resultados para nuestros accionistas.”

- **¿Cómo piensa que va a impactar la aplicación en los sectores de la empresa que comenzarán a utilizarla?**

“En términos de imagen muestra que el equipo de dirección está permanentemente buscando simplificar, pero, a la vez, dignificar las distintas funciones y tareas para que nuestro personal sienta motivación en agregar valor a su trabajo por la vía de análisis e iniciativas vinculadas al funcionamiento del servicio. Desde el punto de vista del personal de operaciones, estas innovaciones brindan cada vez más garantías a la gran mayoría que se desempeña en buena forma a pesar de la hostilidad del medio en que trabaja (tránsito) y desde el punto de vista de los clientes nos sigue mostrando como una empresa innovadora y vanguardista.”

- **¿Le parece que el alcance logrado fue adecuado?**

“Si, naturalmente, a partir de su puesta en producción será inevitable que crezca integrando otras áreas y funciones. Pero en esta instancia el alcance es incluso mayor al imaginado en primera instancia, especialmente por su interacción y complementación con la aplicación que desarrollamos (C.U.T.C.S.A. app/api).”

- **¿Qué opina de la usabilidad de la aplicación?**

“Luego de un entrenamiento mínimo no va a ofrecer dificultades, la interface es muy intuitiva y amigable. Desde el punto de vista estético, está en el nivel esperado en función de los estándares actuales.”

- **En la primera reunión que tuvimos con usted, en la cual le explicamos el objetivo de nuestro proyecto, nos comentó que el mismo posicionaría a C.U.T.C.S.A. "en una situación envidiable frente a otras empresas, incluso en el mundo". ¿Sigue pensando de la misma manera?**

“Sí. Conozco la forma de trabajo y las herramientas de gestión en muchas ciudades del mundo. Este proyecto nos posiciona en vanguardia, especialmente teniendo en cuenta las condiciones de nuestro tránsito. Poder integrar información sobre el estado del mismo y generar alarmas será útil, no solo para nuestro personal y clientes, sino también para el resto de los componentes del sistema (peatones, automovilistas, autoridades, etc.).

Estamos generando una especie de embrión de un Centro de Movilidad y esto también es consecuencia de que estamos presentes en toda el Área Metropolitana.”

- **¿Algún otro comentario que desee hacer?**

“Es muy bueno que el ambiente académico propicie proyectos que tienen alta vinculación con las necesidades concretas de la sociedad y que impactan positivamente en la calidad de vida de la población.”

12.22. Anexo 22 – Análisis de puntos de función

A continuación, se describe el análisis completo de los puntos de función para los requerimientos de cada evolución.

12.22.1. Primera evolución

Tipo	RET	DET	Valor	RF usado
ELF	Usuario	id_usuario, contraseña, rol	5	G-RF1
ILF	Menu	Nombre, id_menu, tipo	7	G-RF3
ILF	Reporte	exception, titulo, descripción, información_adicional_usuario	7	G-RF4

Tabla 12-89 Análisis de ILF y ELF para la primera evolución

EI			
RF	Entrada	FTR	Complejidad
G-RF1: Log In	Inicio de sesión	Usuario	Baja
G-RF2: Log Out	Cierre de sesión	Usuario	Baja
G-RF3: Mostrar Menú	-	-	-
G-RF4: Reportar un error en la aplicación	Reportar error con comentario del usuario	Reporte	Baja
	Reportar error con mensaje predeterminado	Reporte	Baja

Ver mapa	-	-	-
----------	---	---	---

Tabla 12-90 Análisis de EI para la primera evolución

EO			
RF	Entrada	FTR	Complejidad
G-RF1: Log In	-	-	-
G-RF2: Log Out	-	-	-
G-RF3: Mostrar Menú	Mostrar opciones del menú	Menu	Baja
G-RF4: Reportar un error en la aplicación	-	-	-
Ver mapa	Mostrar posición del usuario	-	Baja
	Mostrar un conjunto de marcadores	-	Baja
	Esconder un conjunto de marcadores	-	Baja
	Mostrar información de un marcador	-	Baja
	Centrar un marcador en el mapa	-	Baja

Tabla 12-91 Análisis de EO para la primera evolución

EQ				
RF	Consulta	EI FTR	EO FTR	Complejidad
G-RF1: Log In	-	-	-	-
G-RF2: Log Out	-	-	-	-
G-RF3: Mostrar Menú	-	-	-	-
G-RF4: Reportar un error en la aplicación	-	-	-	-
Ver mapa	-	-	-	-

Tabla 12-92 Análisis de EQ para la primera evolución

RF	Valor ILF	Valor ELF	Valor EI	Valor EO	Valor EQ	Total PF
G-RF1: Log In	0	5	3	0	0	8
G-RF2: Log Out	0	0	3	0	0	3
G-RF3: Mostrar Menú	7	0	0	4	0	11
G-RF4: Reportar un error en la aplicación	7	0	6	0	0	13
Ver mapa	0	0	0	20	0	20

Tabla 12-93 Total de PF para la primera evolución

12.22.2. Segunda evolución

Tipo	RET	DET	Valor	RF usado
ELF	Parada	id_parada, lista(líneas), direccion, posicion	5	C2-RF1: Ver información de paradas
ELF	Linea	nombre_linea, lista(recorridos)	5	C2-RF1: Ver información de paradas
ELF	Recorrido	nombre_recorrido, lista(horarios), lista(paradas), origen, destino, lista(puntos_de_control), sentido	5	C2-RF1: Ver información de paradas
ELF	HorarioRecorrido	hora_salida, hora_llegada	5	C2-RF1: Ver información de paradas
ELF	Coche	numero_coche, posicion	5	C2-RF4: Ubicar coche en un mapa
ILF	Movimiento Ruta	numero, id_ruta, parada, tiempo_de_espera, tiempo_de_viaje	7	C2-RF5: Ver ruta entre un origen y un destino
ELF	PuntoInteres	posición, dirección, teléfono, horario_semana, horario_sabado, horario_dominngo, nombre_punto_interes, barrio	5	C2-RF6: Ver información de puntos de interés
ELF	Servicio	Id_servicio, gr, tm, linea, recorrido, nro_conexion, horario_comienzo, horario_fin, coche	5	C2-RF7: Control del servicio

ILF	Control Servicio	servicio, primer_boleto_vendido, ultimo_boleto_vendido, nro_sam, lista(boletos anulados), coche, funcionario_conductor, funcionario_guarda, nro_liquidacion, nro_basica, lista(tarjetas_viaje)	10	C2-RF7: Control del servicio
ELF	Tarjeta	Id_tarjeta, tipo	5	C2-RF7: Control del servicio
ILF	Boleto	nro_boleto, fecha_vendido, hora_vendido, tipo	7	C2-RF7: Control del servicio
ELF	PuntoControl	posicion, direccion, hora_pasada	5	C2-RF7: Control del servicio

Tabla 12-94 Análisis de ILF y ELF para la segunda evolución

EI			
RF	Entrada	FTR	Complejidad
C2-RF1: Ver información de paradas	-	-	-
C2-RF2: Visualizar el recorrido de cierta línea en un mapa	-	-	-
C2-RF3: Mostrar arribo de líneas a la parada	-	-	-
C2-RF4: Ubicar coche en un mapa	-	-	-
C2-RF5: Ver ruta entre un origen y un destino	-	-	-

C2-RF6: Ver información de puntos de interés	-	-	-
C2-RF7: Control del servicio	-	-	-
C2-RF8: Mostrar horario planificado de una línea	-	-	-

Tabla 12-95 Análisis de EI para la segunda evolución

EO			
RF	Entrada	FTR	Complejidad
C2-RF1: Ver información de paradas	-	-	-
C2-RF2: Visualizar el recorrido de cierta línea en un mapa	-	-	-
C2-RF3: Mostrar arribo de líneas a la parada	-	-	-
C2-RF4: Ubicar coche en un mapa	-	-	-
C2-RF5: Ver ruta entre un origen y un destino	-	-	-
C2-RF6: Ver información de puntos de interés	-	-	-
C2-RF7: Control del servicio	-	-	-
C2-RF8: Mostrar horario planificado de una línea	-	-	-

Tabla 12-96 Análisis de EO para la segunda evolución

EQ				
RF	Consulta	EI FTR	EO FTR	Complejidad
C2-RF1: Ver información de paradas	Buscar dirección	-	Parada, Línea	Baja
	Buscar parada	Parada	Parada, Línea	Baja
	Filtrar línea	Línea	Recorrido	Baja
	Ver parada en el mapa	Parada	-	Baja
	Ver horarios de una línea	Línea	-	Baja
	Reportar error de la parada	ReporteError	-	Baja
	Mostrar recorridos de una línea	Recorrido	Recorrido	Baja
C2-RF2: Visualizar el recorrido de cierta línea en un mapa	Buscar recorrido	Línea, Recorrido	Parada, Línea, Recorrido	Baja
	Ver información de parada	Parada	-	Baja
	Reportar error de recorrido	ReporteError	-	Baja
	Listar paradas	-	Parada, Recorrido	Baja

C2-RF3: Mostrar arribo de líneas a la parada	Buscar parada por número filtrando por línea	Parada, Linea	Parada, Linea, Prediccion	Baja
	Ver recorrido en el mapa	Recorrido	-	Baja
	Ver parada en el mapa	Parada	-	Baja
	Reportar error	ReporteError	-	Baja
	Ver coche en mapa	Recorrido	-	Baja
C2-RF4: Ubicar coche en un mapa	Buscar línea	Linea	Lista(Coche, Servicio)	Baja
	Buscar coche	Coche	Coche, Servicio	Baja
	Reportar error del coche	ReporteError	-	Baja
	Ver coche en el mapa	Servicio	-	Baja
C2-RF5: Ver ruta entre un origen y un destino	Buscar ruta		MovimientoRuta , Parada	Baja
	Ver información de parada	Parada	-	Baja
	Ver ruta en lista	-	-	Baja
	Reportar error de ruta	ReporteError	-	Baja

	Ver en mapa-	-	-	Baja
C2-RF6: Ver información de puntos de interés	Buscar dirección, buscar punto de interés	PuntoInteres	PuntoInteres	Baja
	Realizar llamada telefónica al punto de interés	PuntoInteres	-	Baja
	Reportar error del punto de interés	ReporteError	-	Baja
	Ver ruta para llegar hasta ese punto	PuntoInteres	-	Baja
C2-RF7: Control del servicio	Obtener información del servicio de la tarjeta	ControlServicio, Servicio, Boleo, PuntoDeControl	ControlServicio, Servicio, Boleto, PuntoDeControl	Media
	Reporte error de control	ReporteError	-	Baja
	Controlar las tarjetas de los pasajeros	Tarjeta	ControlServicio	Baja
C2-RF8: Mostrar horario planificado de una línea	Buscar línea filtrada por recorrido para cierta fecha	Linea, Recorrido	Parada, Linra, Recorrido	Baja
	Reportar error de horarios	ReporteError	-	Baja

	Buscar servicio	Servicio	Parada, Línea, Recorrido	Baja
--	-----------------	----------	--------------------------	------

Tabla 12-97 Análisis de EQ para la segunda evolución

RF	Valor ILF	Valor ELF	Valor EI	Valor EO	Valor EQ	Total PF
C2-RF1: Ver información de paradas	0	20	0	0	21	41
C2-RF2: Visualizar el recorrido de cierta línea en un mapa	0	0	0	0	12	12
C2-RF3: Mostrar arribo de líneas a la parada	0	0	0	0	15	15
C2-RF4: Ubicar coche en un mapa	0	25	0	0	12	37
C2-RF5: Ver ruta entre un origen y un destino	7	0	0	0	15	22
C2-RF6: Ver información de puntos de interés	0	5	0	0	12	17
C2-RF7: Control del servicio	17	15	0	0	10	42
C2-RF8: Mostrar horario planificado de una línea	0	0	0	0	9	9

Tabla 12-98 Total de PF para la segunda evolución

12.22.3. Tercera evolución

Tipo	RET	DET	Valor	RF usado
------	-----	-----	-------	----------

ELF	Notificacion	id_notificacion, decripcion, estado, fecha, tipo, importancia	Baja	G-RF5: Recepción de notificaciones
ELF	ServicioAsignado	servicio, lista(usuario_asignados)	Baja	C3-RF1: Consultar servicios asignados
ELF	Noticia	id_noticia, titulo, descripción, imagen, imagen_preview	Baja	G-RF7: Visualización de noticias

Tabla 12-99 Análisis de ILF y ELF para la tercera evolución

EI			
RF	Entrada	FTR	Complejidad
G-RF5: Recepción de notificaciones	-	-	-
G-RF6: Gestión de notificaciones	Eliminar notificación	Notificacion	Baja
	Eliminar todas las notificaciones	Notificacion	Baja
G-RF7: Visualización de noticias	-	-	-
C3-RF1: Consultar servicios asignados	-	-	-
C4-RF1: Modificar servicios, C4-RF2: Alta	Actualizar un viaje	ServicioAsignado	Baja
	Crear nuevo viaje	ServicioAsignado	Baja

servicio, servicio	C4-RF3: Baja	Eliminar un viaje	ServicioAsignado	Baja
--------------------	--------------	-------------------	------------------	-------------

Tabla 12-100 Análisis de EI para la tercera evolución

EO			
RF	Entrada	FTR	Complejidad
G-RF5: Recepción de notificaciones	-	-	-
G-RF6: Gestión de notificaciones	-	-	-
G-RF7: Visualización de noticias	-	-	-
C3-RF1: Consultar servicios asignados	Listar los servicios asignados al usuario	Servicio, Usuario	Baja
	Resaltar el próximo servicio que debe realizar el usuario	Servicio, Usuario	Baja
	Ver detalles del servicio asignado al usuario	Servicio, Usuario	Baja
C4-RF1: Modificar servicios, C4-RF2: Alta servicio, C4-RF3: Baja servicio	-	-	-

Tabla 12-101 Análisis de EO para la tercera evolución

EQ			
RF	Entrada	FTR	Complejidad
G-RF5: Recepción de notificaciones	Recibir y alertar una nueva notificación	Notificación, Usuario	Baja
G-RF6: Gestión de notificaciones	Listar todas las notificaciones	Notificación, Usuario	Baja
	Buscar una notificación por su nombre	Notificación, Usuario	Baja
	Buscar una notificación de cierto tipo	Notificación, Usuario	Baja
	Mostrar cantidad de nuevas notificaciones	Notificación, Usuario	Baja
	Ver detalles de una notificación	Notificación, Usuario	Baja
G-RF7: Visualización de noticias	Listar todas las noticias	Noticias	Baja
C3-RF1: Consultar servicios asignados	-	-	-
C4-RF1: Modificar servicios, C4-RF2: Alta servicio, C4-RF3: Baja servicio	Listar todos los servicios de la línea agrupados por fecha	Servicio, Usuario	Baja
	Ver detalles de un servicio	Servicio, Usuario	Baja

Tabla 12-102 Análisis de EQ para la tercera evolución

RF	Valor ILF	Valor ELF	Valor EI	Valor EO	Valor EQ	Total PF
G-RF5: Recepción de notificaciones	0	5	0	0	3	8
G-RF6: Gestión de notificaciones	0	0	6	0	15	21
G-RF7: Visualización de noticias	0	5	0	0	3	8
C3-RF1: Consultar servicios asignados	0	5	0	12	0	17
C4-RF1: Modificar servicios, C4-RF2: Alta servicio, C4-RF3: Baja servicio	0	0	9	0	6	15

Tabla 12-103 Total de PF para la tercera evolución

12.23. Anexo 23 – Reunión final con *Sponsor*

El día 23 de febrero del 2016, se realizó una reunión con el *Sponsor* del proyecto, Manuel Áres, con el objetivo de describir de forma resumida la funcionalidad completa del producto.

Luego de la reunión, se le consultó su opinión general del proyecto. La respuesta se detalla a continuación.

“Me siento conforme con el proyecto en general, el proceso fue brillante y creemos que se les ofreció toda la ayuda que necesitaron.

Con la aplicación móvil en particular, muy conforme, pero con la ambición que se tiene por querer más. Luego que se presente el proyecto, se va a seguir haciendo cosas porque al mismo le veo mucho futuro. Es muy interesante para la explotación nuestra.

Este es un producto que, para quien no conoce sobre el transporte, es difícil de realizar y entender. Para nosotros son cosas de todos los días, pero para ustedes fue difícil entenderlo. Se deberá realizar un esfuerzo grande para que en la presentación entiendan los términos y funciones que tenemos dentro de la empresa.

Sobre el alcance, en mi opinión fue adecuado, pero se puede mejorar un 40 o 50% de lo que se tiene actualmente para cosas futuras que se piensen realizar.

Es un producto modular que hoy se llegaron a realizar cuatro roles, pero que en un futuro llegarán a ser más.

Con respecto a la seguridad, creemos que cumple satisfactoriamente con el nivel esperado para la empresa. Y sobre la usabilidad, el sistema lo considero muy amigable para el usuario.”
