

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

SISTEMA DE GESTIÓN MÓVIL PARA UNA EMPRESA DE TRANSPORTES

SEBASTIÁN ALDO COSTAGUTA CABRERA

INFORME FINAL DEL PROYECTO
PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO DE EJECUCIÓN EN INFORMÁTICA

DICIEMBRE 2012

PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO
FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA INFORMÁTICA

SISTEMA DE GESTIÓN MÓVIL PARA UNA EMPRESA DE TRANSPORTES

SEBASTIÁN ALDO COSTAGUTA CABRERA

Profesor Guía: **José Rubio León**

Profesor Co-referente: **Broderick Crawford Labrín**

Carrera: **Ingeniería de Ejecución en Informática**

DICIEMBRE 2012

*Dedicado a Renata, mi hija,
que le dio sentido a mi vida y me devolvió las ganas de sonreír.
Dedicado a Macarena, mi novia,
quien me acompañó en todo este proceso, siendo mi compañera y guía
fundamental para superar las adversidades de la vida.
Dedicado a Aldo, Carmen, Bruno.
A ellos les debo lo que soy, esta herencia abierta de valores,
hacen que trate de ser cada día mejor.*

“In bocca al lupo”

Índice

Índice.....	i
Resumen.....	iv
Abstract.....	iv
Lista de Figuras.....	v
Lista de Tablas.....	vii
1 Capítulo primero: Introducción.....	1
2. Capítulo segundo: Estado del arte.....	2
2.1 Necesidad detectada y problemática a resolver.....	2
2.2 Características de dispositivos móviles.....	3
2.2.1 Definición de un dispositivo móvil.....	3
2.3 Tipos de Dispositivos Móviles.....	4
2.4 Tecnologías existentes para el desarrollo del sistema.....	5
2.4.1 Java ME.....	5
2.4.2 Android.....	6
2.4.3 Windows Phone.....	10
2.4.4 SQL Server.....	12
3 Capítulo tercero: Metodología.....	13
3.1 Modelo de Proceso: Iterativo Incremental.....	13
3.2 Objetivos.....	15
3.2.1 Objetivo General.....	15
3.2.2 Objetivos Específicos.....	15
3.2.3 Plan de Trabajo.....	16
3.3 Análisis de Riesgos.....	17
3.3.1 Identificación de riesgos.....	17
3.3.2 Plan de mitigación.....	18
3.3.3 Plan de contingencia.....	19
4 Capítulo cuarto: Análisis.....	20
4.1 Factibilidad Técnica.....	21
4.1.1 Tecnología a utilizar para el desarrollo.....	21
4.2 Requerimientos Funcionales.....	22

4.3	Requerimientos No Funcionales	23
4.4	Casos de Uso	24
4.4.1	Caso de Uso de Alto Nivel	24
4.4.2	C.U: Generar Reportes de Estado.....	25
4.4.3	C.U: Gestionar Reportes.....	32
4.5	Diagramas de Actividad	37
4.5.1	Actividad: Ingreso Operario	37
4.5.2	Actividad: Buscar Operario.....	38
4.5.3	Actividad: Generar Modulo de Trazabilidad	39
4.5.4	Actividad: Buscar Guía Despacho.....	40
4.5.5	Actividad: Enviar Reporte.....	41
4.5.6	Actividad: Desplegar Historial	42
4.5.7	Actividad: Validar Reporte	43
4.5.8	Actividad: Modificar Reporte	44
4.6	Diagramas de Secuencia.....	45
4.6.1	Ingreso de operario carga	45
4.6.2	Generación de reporte	46
4.6.3	Despliegue de Historial.....	47
4.6.4	Validar/Modificar Reporte	48
4.7	Diagrama de Clases.....	49
4.8	Protocolos.....	50
4.8.1	Protocolo a utilizar: Protocolo TCP	50
4.8.2	Servidor HTTP	51
4.8.3	Servicios Web	53
4.8.4	Tecnología a utilizar para la comunicación de datos	55
4.9	Factibilidad Legal.....	56
5	Capítulo quinto: Diseño	57
5.1	Arquitectura del Sistema	57
5.2	Modelo de Datos	58
5.3	Modelo de datos actual (extracto)	59
5.3.1	Modificación al Modelo Relacional actual	60

5.4 Diseño Interfaces.....	61
5.4.1 Interfaz: Ingreso Operario de Carga.....	61
5.4.2 Interfaz: Operario de Carga - Historial Reporte y Pestaña General	62
5.4.3 Interfaz: Operario de Carga - Vista Principal – Pestaña Operario	63
5.4.4 Interfaz: Operario de Carga - Creación Reporte.....	64
5.5 Prototipo Funcional.....	66
5.5.1 Selección Aplicación desde el Menú	66
5.5.2 Ingreso de usuario (Operario de carga)	67
5.5.3 Menú principal	68
5.5.4 Menú principal – Envío de Reporte.....	69
5.5.5 Menú principal – Pestaña Mensaje y opciones.....	70
5.5.6 Menú principal – Pestaña Ayuda.....	71
5.6 Consideraciones para un óptimo desempeño	72
5.7 Definición métodos Servicio Web	73
6 Capítulo sexto: Pruebas.....	74
7 Capítulo séptimo: Conclusiones y Trabajo Futuro.....	75
8 Capítulo octavo: Referencias y Bibliografía	77

Resumen

La habilidad de poder interactuar con la información de la empresa, sumado con la intención de dar más agilidad al rubro del negocio y poder captar información precisa de una operación, ha llevado a un gran número de empresas tanto a nivel nacional como internacional a implementar en sus sistemas o en gran parte de ellos un entorno móvil.

A través de este proyecto, se implementa un software, el que ayuda a mejorar la labor diaria de una gran cantidad de operarios de planta, así como también los encargados de transportar los diversos contenedores de los clientes.

Se implementa el modulo más relevante en la relación de la empresa con el cliente, el cual consta de un servicio de trazabilidad, que es capaz de informar al cliente lo que está sucediendo con su contenedor, es decir, donde se encuentra, que fue lo último que realizó el operario de la carga, en que tramo se encuentra, si existe algún problema con el tráfico vehicular y en el peor de los casos, si existe algún desperfecto técnico.

Palabras clave: Dispositivos Móviles, Trazabilidad, Servicios Web, Android

Abstract

The ability to interact with business information, coupled with the intention to give more flexibility to the line of business and to capture accurate information for an operation, has led to a large number of companies both nationally and internationally to implement systems or a major part of them a mobile environment.

Through this project, we implement a software, which helps improve the daily work of a large number of plant operators and managers also carry containers of various customers.

It implements the most important module in the company's relationship with the client, which includes a tracking service, which is able to inform the customer what is happening with its container, if there is a problem with vehicular traffic and in the worst case, if there is a technical malfunction.

Keywords: Mobile Computing Device, Tracking, Servicio Web, Android

Lista de Figuras

Figura 1, Arquitectura Java ME [3].....	5
Figura 2, Arquitectura Android [4].....	7
Figura 3, Cuota de Versiones de Android [5]	9
Figura 4, Modelo de Software Windows Phone [6].....	10
Figura 5, Ejecución de aplicaciones, limitaciones y Frameworks [7].....	11
Figura 6, Arquitectura SQL Server 2005 [8].....	12
Figura 7, Modelo Iterativo Incremental.....	14
Figura 8, Caso de Uso de Alto Nivel	24
Figura 9, Caso de Uso Generar Reportes de Estado	25
Figura 10, Caso de Uso Ingreso Operario.....	26
Figura 11, Caso de Uso Buscar Operario.....	27
Figura 12, Caso de Uso Generar Modulo de Trazabilidad.....	28
Figura 13, Caso de Uso Buscar Guía Despacho.....	29
Figura 14, Caso de Uso Enviar Reporte.....	30
Figura 15, Caso de Uso Desplegar Historial.....	31
Figura 16, Caso de Uso Gestionar Reporte.....	32
Figura 17, Caso de Uso Validar Reporte.....	33
Figura 18, Caso de Uso Modificar Reporte.....	34
Figura 19, Caso de Uso Guardar Reporte.....	35
Figura 20, Caso de Uso Reporte Cliente	36
Figura 21, Actividad Ingreso Operario	37
Figura 22, Actividad Buscar Operario	38
Figura 23, Actividad Generar Modulo Trazabilidad.....	39
Figura 24, Actividad Buscar Guía Despacho.....	40
Figura 25, Actividad Enviar Reporte	41
Figura 26, Actividad Desplegar Historial	42
Figura 27, Actividad Validar Reporte	43
Figura 28, Actividad Modifica Reporte	44
Figura 29, Secuencia Ingreso de Operario Carga	45
Figura 30, Secuencia Generación de Reporte.....	46
Figura 31, Secuencia Despliegue Historial.....	47
Figura 32, Secuencia Validar/Modificar Reporte	48
Figura 33, Diagrama de Clases	49
Figura 34, Transferencia en el protocolo TCP.....	50
Figura 35, Arquitectura de Servidor HTTP [11].....	51
Figura 36, Caso de Uso de una Petición HTTP.....	52
Figura 37, Arquitectura Servicio Web.....	54
Figura 38, SOAP y los mensajes XML [14].....	55
Figura 39, Arquitectura del Sistema.....	57
Figura 40, Modelo de datos propuesto.....	58
Figura 41, Modelo Relacional actual del ERP.....	59
Figura 42, Relación de la nueva tabla Posición reporte.....	60
Figura 43, Relación de la nueva tabla Operario carga.....	60

Figura 44, Interfaz de Ingreso y validación.....	61
Figura 45, Interfaz Historial Reporte y Pestaña General.....	62
Figura 46, Interfaz Vista General Pestaña Operario.....	63
Figura 47, Interfaz Creación Reporte.....	64
Figura 48, Interfaz Creación Reporte Validación.....	65
Figura 49, Selección aplicación.....	66
Figura 50, Prototipo Ingreso de usuario.....	67
Figura 51, Prototipo Menú principal.....	68
Figura 52, Prototipo Menú principal – Envío reporte.....	69
Figura 53, Prototipo Menú principal – Mensaje y opciones.....	70
Figura 54, Prototipo Menú principal – Ayuda.....	71

Lista de Tablas

Tabla 1, Plan de Trabajo.....	16
Tabla 2, Identificación Riesgos.....	17
Tabla 3, Mitigación.....	18
Tabla 4, Plan Contingencia.....	19
Tabla 5, Requerimientos Funcionales.....	22
Tabla 6, Caso de uso Ingresar datos	26
Tabla 7, Caso de uso Validar datos	26
Tabla 8, Caso de uso Validar Operario	27
Tabla 9, Caso de uso Obtener datos Operario	27
Tabla 10, Caso de uso Desplegar datos Operario	27
Tabla 11, Caso de uso Generar módulo de Trazabilidad	28
Tabla 12, Caso de uso Desplegar Sugerencias.....	28
Tabla 13, Caso de uso Ingresar Información Reporte	28
Tabla 14, Caso de uso Validar Información del Reporte	29
Tabla 15, Caso de uso Buscar Guía Despacho	29
Tabla 16, Caso de uso Obtener datos Guía Despacho	29
Tabla 17, Caso de uso Desplegar datos Guía Despacho	30
Tabla 18, Caso de uso Validar Datos del Reporte	30
Tabla 19, Caso de uso Enviar Datos al Operario	30
Tabla 20, Caso de uso Desplegar Información Básica	31
Tabla 21, Caso de uso Actualizar Historial	31
Tabla 22, Caso de uso Generar Estructura Reporte	33
Tabla 23, Caso de uso Ingresar Información nueva	34
Tabla 24, Caso de uso Generar Estructura del Reporte	34
Tabla 25, Caso de uso Actualizar Historial Reportes... ..	35
Tabla 26, Caso de uso Obtener datos Cliente	36
Tabla 27, Caso de uso Validar Datos Cliente	36
Tabla 28, Caso de uso Enviar Reporte Email	36
Tabla 29, Optimización recursos.....	72

1 Capítulo primero: Introducción

Innovar es la forma de diferenciarse de la competencia y por sobre todo de dar un valor agregado al trabajo que se entrega a diario a los clientes.

Por medio de este proyecto, se pretende no sólo innovar, sino que además poder ofrecer una solución estable y llamativa, que sea la encargada de operar en un proceso muy importante de la empresa, el cual es la trazabilidad de los servicios.

El proyecto, busca implementar un entorno móvil de fácil utilización, es por eso que se basa en estudios de usabilidad móvil, pensando que el usuario final no será una persona que está muy familiarizada con los teléfonos móviles, como tampoco con aplicaciones asociadas a ellas. Todo esto generará un gran impacto para la empresa, por lo que el proyecto pretende adaptarse de la forma más simple a las reglas de negocio actuales de la compañía.

Se desarrolla un software capaz de ser utilizado en teléfonos de última generación, así como también aquellos que dispongan de un hardware de gamma más baja.

Además de la aplicación móvil, debemos generar un pequeño sistema encargado de insertar y/o modificar los mensajes que envían los operarios de carga para que el cliente pueda obtener la información referente a su carga sin necesidad de entrar al sistema web actual de la empresa.

Si bien se espera que la cantidad de mensajes modificados por los operarios de planta sea la mínima, el sistema web presentará ciertas normas de seguridad a nivel web, para asegurar la protección de la transferencia de datos, así como también un sistema amigable de fácil utilización.

A su vez, se implementa un Servicio Web el cual será el encargado de gestionar la información que se obtiene desde cada uno de los operarios de carga, como también el ingreso a la aplicación, y las consultas asociadas a la base de datos.

El primer capítulo del documento realiza una breve presentación de lo que se desarrolla. Posterior a esto, el capítulo dos detalla las tecnologías estudiadas y analizadas para llevar a cabo el proyecto. Más adelante, se encuentra la Metodología empleada, la cual especifica el marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo. El capítulo cuarto resume el análisis realizado en todas las reuniones con el cliente las cuales fueron diseñadas en el capítulo quinto. En el sexto capítulo se detallan las pruebas realizadas a los servicios principales de la aplicación. Finalmente, el capítulo séptimo engloba las conclusiones realizadas y los posibles trabajos futuros. Para finalizar el documento, se resumen las fuentes consultadas para la realización del proyecto en todas sus etapas.

2. Capítulo segundo: Estado del arte

2.1 Necesidad detectada y problemática a resolver

El presente proyecto, busca resolver una problemática presente en la empresa SertChile Limitada, la cual se encarga de ofrecer servicios de transporte de contenedores, transporte local y de la coordinación de operaciones multimodales en los principales puertos del país.

Actualmente, poseen un software desarrollado a la medida, el cual incluye un Módulo de RRHH, Modulo de Contabilidad, Modulo de Gestión y Modulo de Trazabilidad.

La problemática surge posterior al despacho del operador de carga, debido a que cada operador de planta, cada una o dos horas, debe ponerse en contacto con el operador de carga, para obtener información referente a las tareas diarias que debe realizar, como lo son ir en búsqueda de la carga, realizar el cálculo del peso, los traslados que se realizan, la llegada al puerto y la entrega de la carga al navío.

Esto se vuelve caótico, debido a que normalmente se pueden llegar a manejar entre 60 a 70 camiones, por lo que la mayoría de las veces los operadores de planta no dan el abasto necesario, siendo incapaces de realizar todas las llamadas en los tiempos establecidos, lo que genera que el cliente no pueda obtener la información relativa a su carga en los tiempos estimados, consecuencia de esto se debe estar constantemente llamando a la central de la empresa para saber qué es lo que sucede con su contenedor o carga.

Implementar un GPS podría eliminar, en cierta medida, la mayoría de los problemas presentados anteriormente, pero la problemática va más allá, ya que aunque se posea un sistema de GPS, se debe recurrir de igual manera a las llamadas telefónicas a los operarios de carga, quedando como prioridad disminuir al mínimo los costos asociados a las llamadas telefónicas entre operarios.

El software a desarrollar en el dispositivo móvil, pretende que automáticamente, cada cierto periodo de tiempo, se emitan notificaciones al operario de carga, para que este sea el responsable de informar su situación actual, comentarios o problemas de diversos tipos.

De igual manera, el sistema buscará para el día actual, la última guía válida para el operario de carga, lo cual desligara al usuario de la mejor forma de la aplicación móvil, siendo más práctico y fácil de usar.

Además, luego de reportar un mensaje, este llegará automáticamente al correo electrónico del operario de planta, siendo este último el encargado de validar el mensaje en caso de que exista algún tipo de error ortográfico o de redacción, para que posteriormente sea insertado a la base de datos visible para el cliente.

2.2 Características de dispositivos móviles

2.2.1 Definición de un dispositivo móvil

Todo dispositivo móvil representa una nueva forma de poder acceder a la información de los sistemas, aunque es necesario definir claramente: **¿Qué es un Dispositivo Móvil?**

Básicamente, un dispositivo móvil puede ser definido como un aparato de un tamaño pequeño con capacidades de procesamiento, conexión a algún tipo de red, memoria de almacenamiento limitada, diseñados para una función específica a pesar de que en la mayoría de los casos es capaz de realizar otras funciones mucho más generales. Si nos basamos en esta definición, existe un gran número de dispositivos móviles, desde los GPS, los teléfonos móviles, los PDAs, Tablet de PC, reproductores de audio, etc.

Para este proyecto, nos centraremos principalmente en los teléfonos debido a que son los más utilizados y conocidos en la actualidad, además de ser los que ofrecen mayor variedad de aplicaciones multimedia y presentan una evolución mucho más amplia.

Las principales características que hacen que estos dispositivos se diferencien de los ordenadores comunes es:

- La funcionalidad está limitada.
- No necesariamente son actualizables (aunque varios si posean esta cualidad).
- Su manejo es mucho más simple y suelen tener una mejor aceptación por parte del usuario final.
- Cada cierta cantidad de tiempo deben ser cambiados.
- Debido a que su manejo es simple, sus opciones son fáciles de aprender.
- No es necesario que el usuario final que va a utilizar el dispositivo móvil sea un usuario experto.

Existen varias categorías de dispositivos móviles debido al variado número de niveles funcionalidad, podemos encontrar:

- Dispositivo móvil de datos limitado: Por lo general poseen una pantalla muy pequeña, basada en pantallas de tipo texto con servicios de datos limitados a SMS y WAP.
- Dispositivo móvil de datos básico: La pantalla suele ser más grande que los anteriores, presentan una navegación basada en iconos los cuales son operados a través de un cursor o rueda y por lo general suelen ofrecer un navegador web, SMS, lista de direcciones, acceso a correo electrónico, etc. Un típico dispositivo que pertenece a esta categoría son los Teléfonos Inteligentes y BlackBerry.
- Dispositivo móvil de datos mejorados: Dispositivos con pantallas medianas a grandes, ofrecen aplicaciones nativas, navegación por medio de *stylus* y aplicaciones de tipo corporativas como SAP, intranet, etc.

2.3 Tipos de Dispositivos Móviles

Podemos definir algunos tipos de dispositivos móviles, los cuales siguen una tendencia de clasificación basada en sus cualidades y funciones.

- Teléfonos móviles pequeños:
 1. Pantallas pequeñas con calidad baja.
 2. Sin reproductor de videos ni MP3.
 3. Posee conexión a internet por una sola vía.
 4. No posee conexiones inalámbricas como WiFi o Bluetooth.
 5. No incluye cámara fotográfica.

- Teléfonos de móviles medianos:
 1. Su capacidad es intermedia
 2. Posee cámara fotográfica, como también puede reproducir MP3 y videos en ciertos formatos livianos.
 3. Incluye pantallas de calidad alta, aunque son consideradas de un tamaño mediano.
 4. Puede establecer una conexión a internet mediante la red 3G y bluetooth.

- Smartphone:
 1. Posee cámara fotográfica, reproduce videos y MP3.
 2. Su pantalla es bastante grande y de una calidad alta.
 3. Se caracteriza por tener una gran cantidad de funcionalidades.
 4. Utiliza un sistema operativo que le permite utilizar aplicaciones de tipo ofimática.
 5. Posee conexión a internet a través de conexiones inalámbricas como WiFi, bluetooth y 3G.

Debido a que se debe cumplir con requerimientos funcionales especiales, como también asegurar la estabilidad, escalabilidad y disponibilidad no es posible utilizar *Teléfonos móviles pequeños* como tampoco *Teléfonos móviles medianos*, debido a que no utilizan un sistema operativo que permitan realizar ciertas tareas específicas para desarrollar lo anteriormente descrito.

2.4 Tecnologías existentes para el desarrollo del sistema

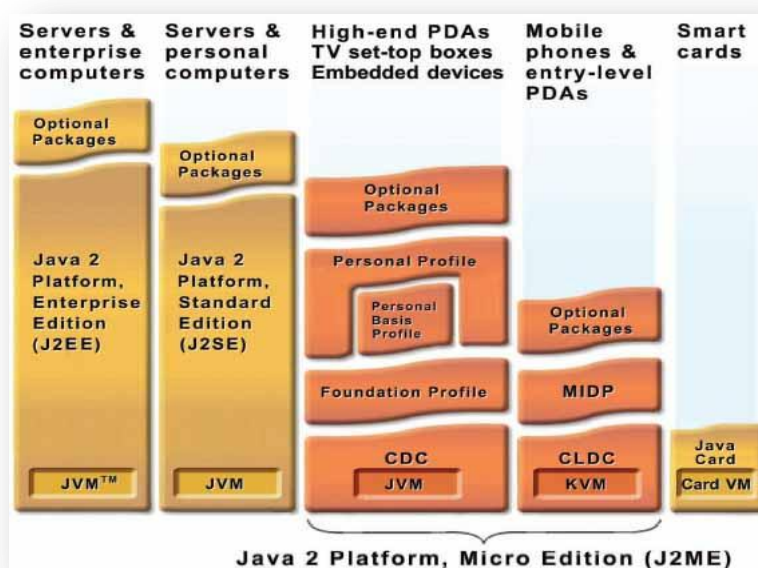
2.4.1 Java ME

Es una plataforma que pertenece al subconjunto de la plataforma java, orientada a ofrecer una colección de APIs para el desarrollo de software con dispositivos de recursos restringidos. Su orientación es hacia los productos de consumo como los PDAs, electrodomésticos o teléfonos móviles.

Java Micro Edition se ha convertido en una opción interesante a la hora de desarrollar juegos en teléfonos móviles, debido a que se puede emular en un PC su funcionamiento (fase de desarrollo) para posteriormente subirlos al teléfono.

La arquitectura se puede dividir en dos grandes bloques de arquitectura, que dependen del tipo de dispositivo y las características de los mismos (**Ver Figura 1**).

Figura 1, Arquitectura Java ME [3]



Para poder tener un entorno que cumpla los requisitos y objetivos de mercado, es necesario que se componga de configuración, perfiles y paquetes opcionales. Todas estas combinaciones, ayudaran a optimizar la memoria, potenciar los procesos y las capacidades de E/S de una categoría de dispositivos.

Las configuraciones están compuestas de una máquina virtual y un conjunto mínimo de bibliotecas de función. Actualmente, existen 2 configuraciones, por un lado tenemos CLDC (Connected Limited Device Configuration) que está diseñada para dispositivos con conexiones de red intermitentes, procesadores lentos, con CPUs de 16 a 32 bits y memoria limitada.

2.4.2 Android

Está enfocado para ser utilizado en dispositivos móviles como: teléfonos inteligentes, Tablet y otros dispositivos.

La cantidad de unidades de teléfonos inteligentes con Android, se ubican en el primer puesto en los Estados Unidos, en el segundo y tercer trimestre del año 2010¹. A nivel mundial alcanzo una cuota de mercado del 50,9% durante el cuarto trimestre de 2011, más que el doble que el segundo sistema más utilizado iOS de iPhone².

Existe una comunidad actual de desarrolladores escribiendo aplicaciones para extender la funcionalidad de los dispositivos. Actualmente, existen más de 400.000 aplicaciones (de ellas, dos tercios son gratuitas) disponibles en las tiendas oficiales de Android.

Las aplicaciones son escritas en Java y C, mientras que su núcleo es de tipo monolítico.

La estructura se compone de aplicaciones que se ejecutan en un framework Java de aplicaciones orientadas a objetos por sobre el núcleo de las bibliotecas de una máquina virtual que posee una compilación en tiempo de ejecución. La bibliotecas escritas en lenguaje C, incluyen un administrador de interfaz gráfica, un framework OpenCore, una base de datos Relacional SQLite, una API grafica de OpenGL ES 2.0 3D y un motor renderizado WebKit.

Dentro de las características y especificaciones actuales, tenemos por un lado el diseño del dispositivo, que es adaptable a pantallas grandes, VGA, etc.

El almacenamiento, está orientado a una base de datos liviana, que es usada para propósitos de almacenamiento de datos (SQLite).

La conectividad se basa en el soporte de tecnologías GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, WiMAX.

El soporte multimedia utiliza los siguientes formatos multimedia: WebM, H263, H264 (3GP o MP4), MPEG-4 SP, AAC, HE-AAC, MP3, MIDI, OggVorbis, WAV, JPEG, PNG, GIF, BMP.

Posee soporte para streaming RTP/RTSP, descarga progresiva de HTML. Posee soporte para hardware adicional, como lo son las cámaras de fotos, de video, GPS, acelerómetros, giroscopios, magnetómetros, sensores de proximidad y de presión, aceleración 2D y 3D.

Incluye soporte para bluetooth con A2DF y AVRCP, envío de archivos (OOP) junto con soporte a Video llamada, utilizando Google Talk.

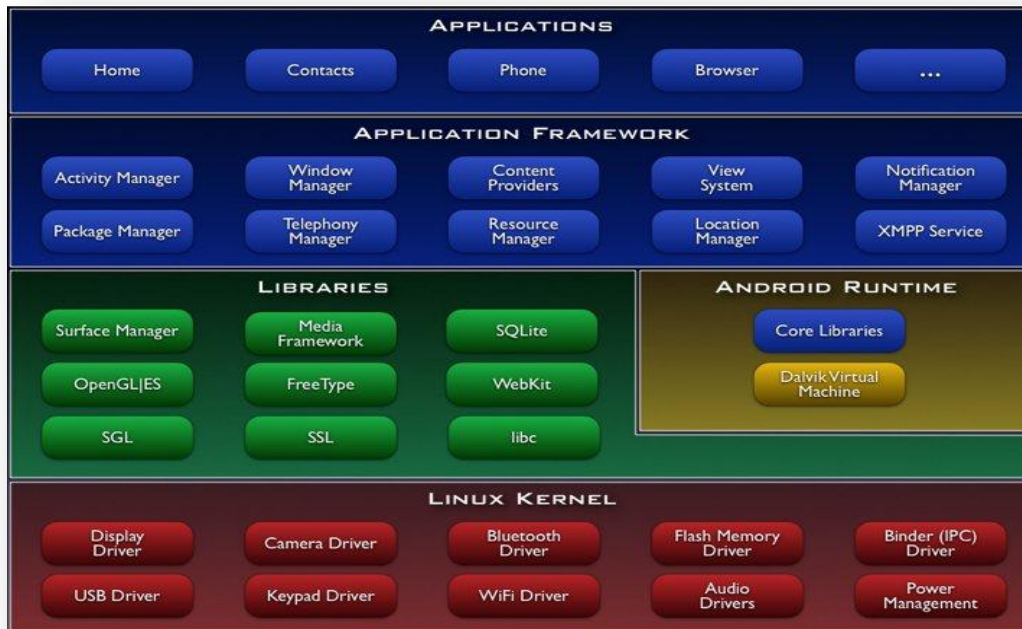
¹Google's Android leads U.S. Smartphones

²Gartner Says Worldwide Smartphone sales solared in fourth Quarte of 2011 with 47 percent growth

2.4.2.1 Arquitectura Android

La descripción general de la arquitectura, puede verse en la **Figura 2**.

Figura 2, Arquitectura Android [4]



Los componentes más relevantes de la arquitectura son:

- Las aplicaciones bases incluyen un cliente de correo electrónico, programa de SMS, calendario, navegador, mapas, contactos y otros. Cabe agregar que todas las aplicaciones están escritas en Java.
- En el Marco de trabajo de aplicaciones los desarrolladores tienen acceso completo a las mismas APIs del framework utilizados por las aplicaciones bases. El diseño está pensado para simplificar la reutilización de componen. Este mismo mecanismo permite que los componentes sean remplazados por el usuario
- Se incluye un conjunto de Bibliotecas de C/C++ que son usadas por una gran cantidad de componentes del sistema, entre ellas podemos nombrar a “System C Library”, bibliotecas de medios, de gráficos 3D y de SQLite.
- Existe un **Runtime de Android** en donde se agrega un set de bibliotecas base que proporcionan la mayoría de las funciones disponibles en las bibliotecas base del lenguaje Java. Cada una de las aplicaciones Android, corre su propio proceso, con su propia instancia en la máquina virtual Dalvik. Éste último, ha sido desarrollado de

forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente, además de estar optimizado para memoria mínima. La máquina virtual está basada en registros y corre clases compiladas por el compilador de Java.

- **Núcleo Linux:** Android es dependiente de Linux para los servicios básicos de seguridad, gestión de memoria, gestión de procesos, pila de red y modelo de controladores. Además, el núcleo tiene la función de actuar como una capa de abstracción entre el hardware y el resto de la pila del software

2.4.2.2 Versiones de Android

La cuota de versiones de uso de Android se muestra en la **Figura 3**

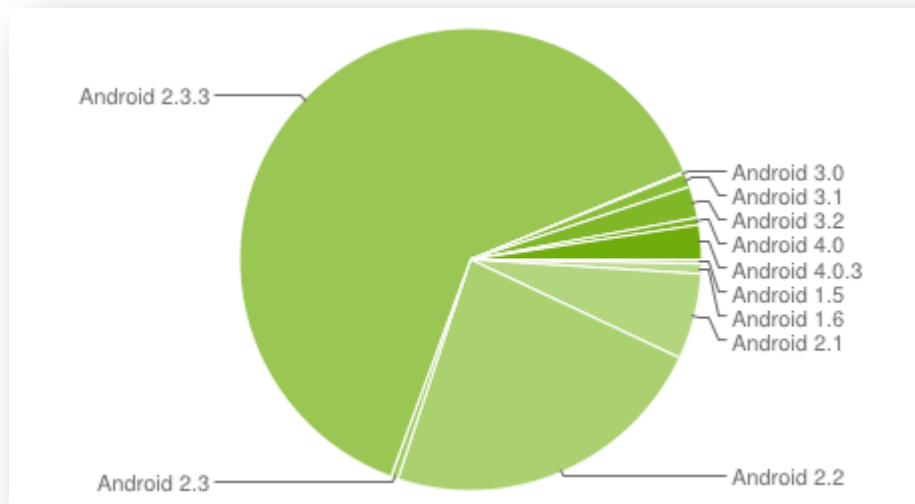


Figura 3, Cuota de versiones de Android [5]

La última versión liberada y estable es la 4.0 conocida como **Ice Cream Sandwich**, la cual posee una interfaz mucho más limpia y moderna con opción de utilizar botones virtuales en la interfaz de usuario en vez de botones táctiles capacitivos. Agrega una aceleración por hardware, lo que produce que la interfaz podrá ser manejada y dibujada por la GPU y aumentando considerablemente su rapidez, su respuesta y por supuesto, la experiencia con el usuario final.

Las tareas pueden ser finalizadas simplemente con su desplazo fuera de la lista.

Se añadió un nuevo gestor de tráfico de datos, que informa mediante alertas cuando se llega a una cuota de uso y la desactivación de datos cuando se excede de su límite.

También se considera corrector de texto el cual fue rediseñado y mejorado, ofreciendo diferentes opciones de edición y sugerencias de palabras similares.

Reconocimiento de voz del usuario, soporte nativo del contenedor MKV y del uso de lápiz táctil.

2.4.3 Windows Phone

Windows Phone 7 está basado en Windows CE 6.0 R3, un sistema en sí mucho más moderno y avanzado que el utilizado en los Windows Mobile 6.x (Windows CE 5.2).

El cambio más relevante de Windows CE 6.0 fue que soporta 43.768 procesos, mientras que la versión anterior sólo soportaba 32 procesos en su Kernel.

La Shell y la plataforma de aplicaciones se almacenan en la memoria del usuario, mientras que el sistema de archivos, networking, los drivers, el sistema de rendering y los gráficos (incluyendo el sistema de actualizaciones residen en el espacio de Kernel.

Actualmente, el sistema es de 32Bits, con lo que sólo es posible manejar 4GB de memoria:

1. 2 GB para procesos.
2. 2 GB para el kernel.

En la versión R3 de Windows CE 6.0 se añadió el soporte para Silverlight Mobile, Internet Explorer Embedded y otras tecnologías. Lo interesante de la versión es el soporte a Flash Lite de forma nativa, a pesar de que esta tecnología por el momento no está disponible en la versión 7 de Windows Phone 7.

El modelo de software (**Ver Figura 4**) se basa en un modelo conciso, pensado para facilitar el uso del dispositivo y aportar cohesión a la interfaz de usuario.

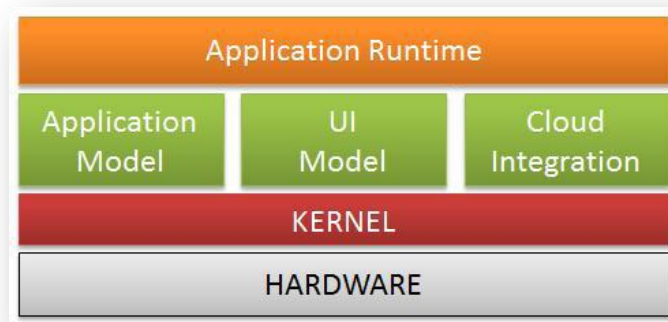


Figura 4, Modelo de software Windows Phone [6]

En el modelo UI (modelo de interfaz de usuario) se incorpora todos los elementos, páginas y sesiones. Cada elemento es considerado un control que se va a mostrar al usuario, toda página es una agrupación de elementos y una sesión es el conjunto de interacciones que se van realizando por un usuario en la aplicación y en algunos casos involucrar a otras aplicaciones.

La forma en la cual se ejecutan las aplicaciones y las limitaciones, junto con los frameworks disponibles, se pueden observar en la **Figura 5**.



Figura 5, Ejecución de aplicaciones, limitaciones y frameworks [7]

Existen dos frameworks disponibles, el “Silverlight for Windows Phone”, el cual nos permite crear aplicaciones con un contenido multimedia amplio que se ejecutan de forma nativa utilizando una interface creada en XAML.

Por otro lado, tenemos “XNA for Windows Phone”, que se basa en XNA, la cual nos permite tener una solución *Multiscreen* (Windows, Xbox, etc) en 2D y 3D para crear juegos con una gran calidad profesional.

Ambos frameworks son ejecutados en un “Sandbox de .NET” que facilita el acceso al hardware, sensores, almacenamiento y localización. Esto nos entrega la opción de que nuestras aplicaciones jamás tendrán acceso nativo al sistema y su ejecución será aislada del sistema.

2.4.4 SQL Server

Es un sistema de gestión para bases de datos relacionales producido por Microsoft que permite lenguajes para consultas T-SQL y ANSI SQL.

Entre sus características más relevantes tenemos:

- Soporte de Transacciones
- Escalabilidad, estabilidad y seguridad
- Soporte de procedimientos almacenados
- Potente entorno gráfico de administración, que permite el uso de comandos DDL y DML.
- Incluye un modo de trabajo cliente-servidor, en el cual la información y los datos son alojados en el servidor y los terminales. Los clientes de la red sólo acceden a la información.

La arquitectura de SQL Server, posee un “Database Engine” el cual permite el almacenamiento, procesamiento y seguridad de los datos. Posee un Analysis Services, para trabajar con datos desde Data Marts, Datawarehouse, y bases de datos de producción/operación. También posee “Integration Services” en el cual se incorporan componentes de tipo ETL (extracción, transformación y carga de datos). A través de “Reporting Services” se permite el almacenamiento de reportes, control de acceso y generación de manera pre-programada. **Ver Figura 6.**

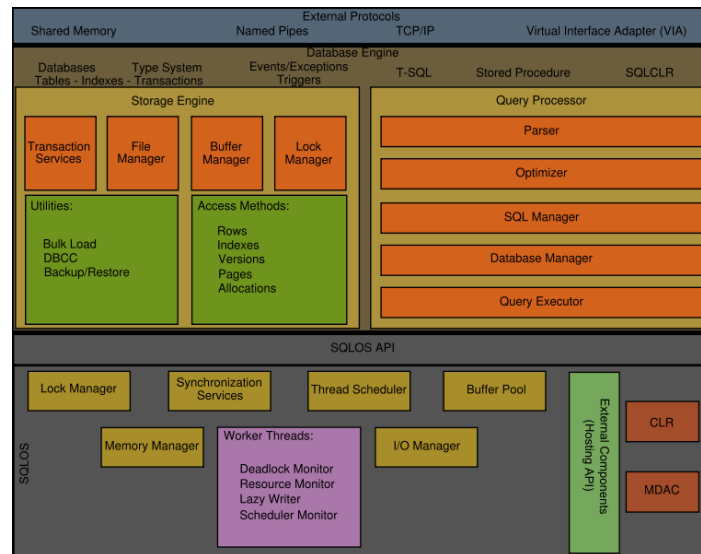


Figura 6, Arquitectura SQL Server 2005 [8]

3 Capítulo tercero: Metodología

3.1 Modelo de Proceso: Iterativo Incremental

Para la implementación del desarrollo de la aplicación móvil que se propone, se han planteado y analizado varias metodologías. Basándose en el modelo que fuera más acorde a los requerimientos como también evaluando la manera de obtener un resultado lo más eficiente y cercano al óptimo, se utiliza la opción que nos permita ir evolucionando a partir de un prototipo (estructura) inicial, con el cual se puedan ir realizando pruebas de su funcionamiento, incluyendo la mayor cantidad de especificaciones posibles, para así disminuir al máximo los errores que puedan existir en las interpretaciones de los requerimientos.

Una de las ventajas, del modelo Iterativo Incremental, es que al ser un modelo híbrido del modelo evolutivo y el de cascada, permite obtener un análisis y diseño mucho más profundo, con avances bien definidos y estructuras más claras, como también permite ir generando los incrementos que serán los elementos específicos del sistema que se irán agregando a medida que se avanza en lo planificado en la fase de análisis y diseño.

Con este modelo, se entregan partes más pequeñas (pero funcionales) las cuales pueden ser priorizadas, lo cual permite acercarse mucho más al cliente y que este pueda ir probando las funcionalidades para así ir mejorando la dependencia entre cada uno de los incrementos (**Ver Figura 7**).

Resulta útil el enfoque de este modelo para el proyecto, debido a que se dispone de sólo una persona para el desarrollo, como también para poder entregar versiones de evaluación.

Este paradigma incremental reduce el tiempo de desarrollo inicial debido a que se va implementando las funcionalidades parcialmente, además de entregar un impacto muy ventajoso frente al cliente y ser muy flexible durante el desarrollo al permitir cambios en los requisitos por parte del cliente, los cuales son propuestos y aprobados para ver dónde y cuándo serán implementados, aunque eventualmente, se debe evaluar la factibilidad de realizar algún cambio porque esto puede generar muchos cambios en las fases anteriores del software, las cuales podrían aumentar en gran medida los costos.

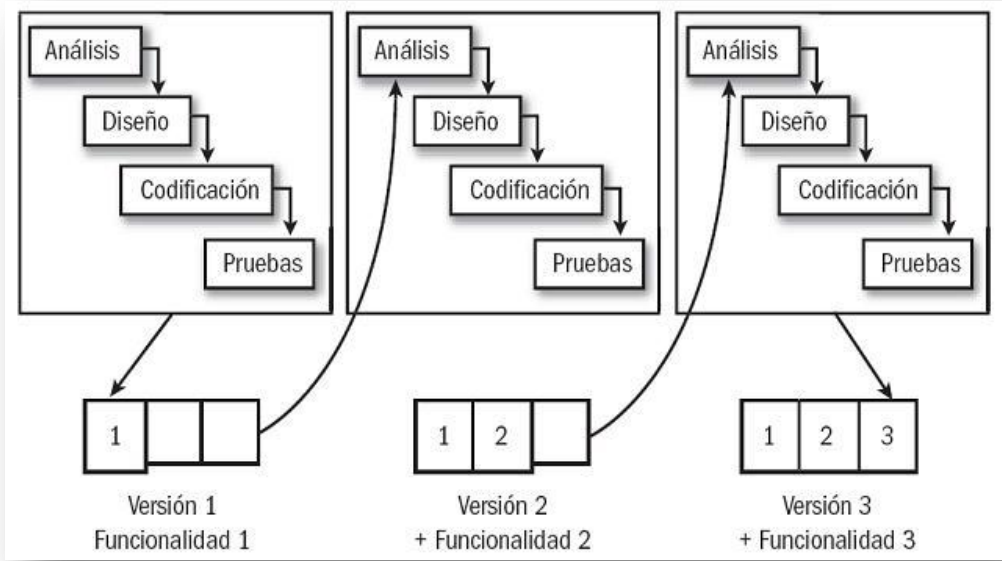


Figura 7, Modelo Iterativo Incremental

3.2 Objetivos

3.2.1 Objetivo General

- Desarrollar un sistema de trazabilidad de camiones en tiempo real para dispositivos móviles que optimice la labor diaria de los operarios de planta de una empresa de transporte terrestre.

3.2.2 Objetivos Específicos

- Describir el estado del arte de las arquitecturas de los sistemas en los cuales se va a implementar la solución, como también la tecnología actual disponible.
- Obtener y analizar los requerimientos del usuario y refinarlos, proponiendo una solución simple y factible en un entorno móvil.
- Diseñar un software capaz de recibir notificaciones diarias, como también de establecer una comunicación con un Servicio Web para poder enviar y recibir información.
- Implementar una aplicación web capaz de gestionar los reportes generados por los operarios, de manera que sea posible validarlos.
- Alimentar un sistema web externo, proporcionando datos aún más exactos que los manejados actualmente por el sistema.

3.2.3 Plan de Trabajo

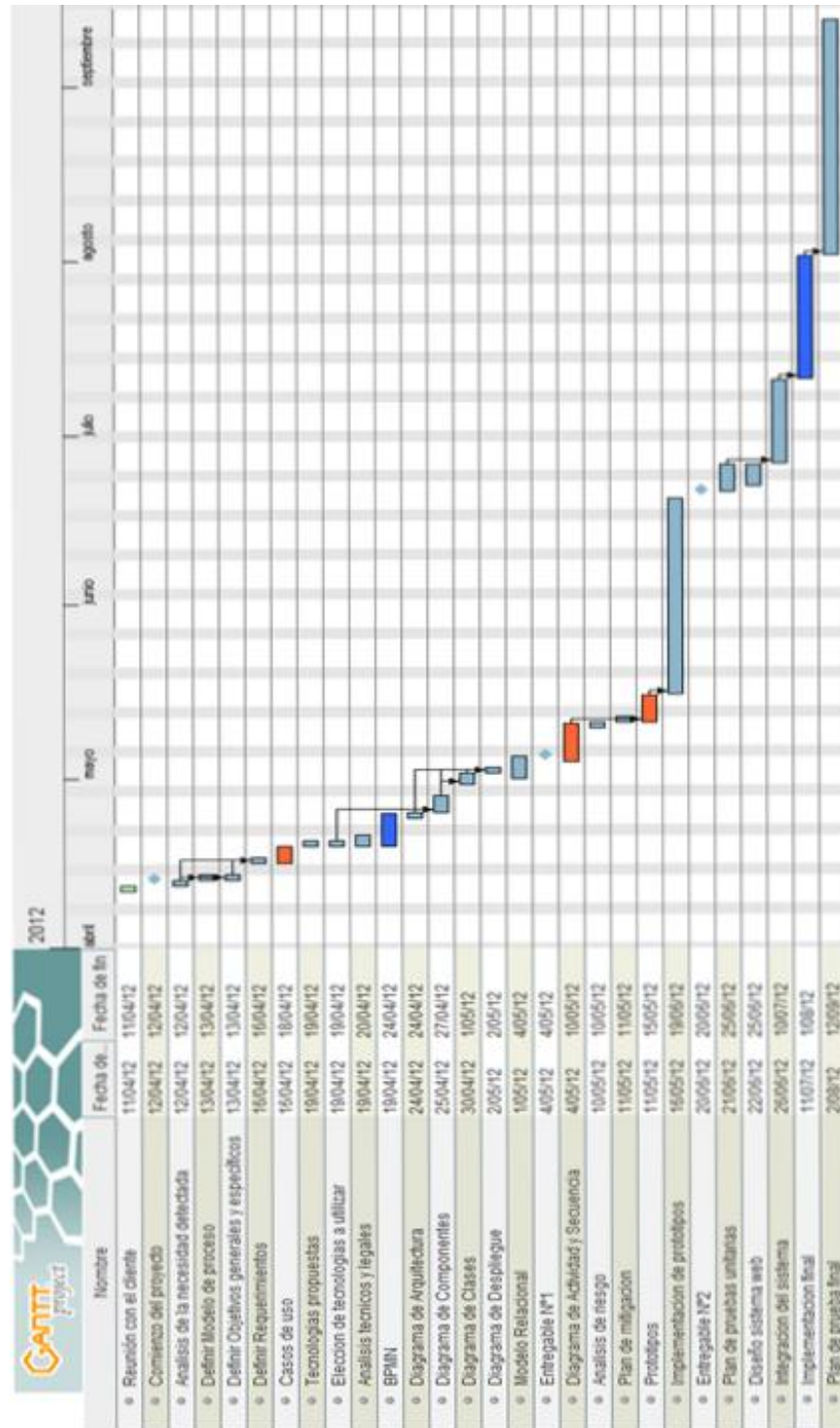


Tabla 1, Plan de Trabajo

3.3 Análisis de Riesgos

Para poder manejar los riesgos e implementar un plan de trabajo es necesario identificar los riesgos, posteriormente evaluarlos y buscar la manera posible de mitigarlos creándose así un plan de trabajo controlado.

3.3.1 Identificación de riesgos

Riesgo	Probabilidad	Efecto
Mala estimación en el tiempo necesario para completar con las tareas fundamentales del Software	Media	Serio
El desarrollador no es capaz de adquirir los conocimientos de los lenguajes involucrado en la aplicación	Alta	Catastrófico
El desarrollador no es capaz de comprender el diseño de interfaz para su desarrollo	Media	Tolerable
Cambios de requisitos de la aplicación por parte del cliente antes de la entrega	Media	Serio
Error en la captura de requerimientos después de la entrega del software	Baja	Catastrófico
Fecha de entrega del software no cumplida	Alta	Serio
Documentación incompleta o no entendible por parte del cliente	Media	Serio
No hay coordinación entre los integrantes del proyecto	Media	Serio

Tabla 2, Identificación Riesgos

3.3.2 Plan de mitigación

Riesgo	Mitigación
Mala estimación en el tiempo necesario para completar con las tareas fundamentales del Software	Generar un plan de trabajo acorde a las exigencias del proyecto y límites pre-establecidos de tiempo
El desarrollador no es capaz de adquirir los conocimientos de los lenguajes involucrado en la aplicación	Se da un tiempo prudente para aprender el/los lenguajes para poder desarrollar la aplicación
El desarrollador no es capaz de comprender el diseño de interfaz para su desarrollo	Se deben generar documentación de diseño claro y entendible. Esta tarea se entrega a personas con la competencia necesaria para realizar esta actividad
Cambios de requisitos del producto software por parte cliente antes de la entrega	Se debe acordar reuniones con los implicados en el diseño de interfaz y personal encargado de requerimientos para evaluar los cambios si es que los hay.
Error en la captura de requerimientos después de la entrega del software	Se revisará en conjunto con el personal implicado
Fecha de entrega no cumplida	Estimar tiempos prudentes para las actividades y planificación personal
Documentación incompleta o no entendible por parte del cliente	Se generará documentación en el proceso de desarrollo del software. Cada proceso del desarrollo tendrá su documentación correspondiente
No hay coordinación entre los integrantes del proyecto	Se establecerán reuniones cuando se estime conveniente.

Tabla 3, Mitigación

3.3.3 Plan de contingencia

Riesgo	Contingencia
Mala estimación en el tiempo necesario para completar con las tareas fundamentales del Software	Reestructuración del tiempo de las tareas de acuerdo a las competencias del alumno
El desarrollador no es capaz de adquirir los conocimientos de los lenguajes involucrado en la aplicación	Se pondrá personal en el grupo de desarrollo con las capacidades de incorporarse y de aprender los lenguajes escogidos
El desarrollador no es capaz de comprender el diseño de interfaz para su desarrollo	Se deben establecer reuniones con los diseñadores de interfaz
Cambios de requisitos del producto software por parte cliente antes de la entrega	Realizar un estudio para ver la factibilidad del cambio en la incorporación del sistema.
Error en la captura de requerimientos después de la entrega del software	Ajustarse a acuerdos con el cliente.
Fecha de entrega no cumplida	No aplica
Documentación incompleta o no entendible por parte del cliente	Adecuar la documentación.
No hay coordinación entre los integrantes del proyecto	Fijar más reuniones presenciales o reuniones online

Tabla 4, Plan Contingencia

4 Capítulo cuarto: Análisis

Para poder llevar a cabo los objetivos, es necesario tener en claro que el software a desarrollar contara con la restricción de que los datos serán obtenidos y cargados por una sola de base de datos relacional, la cual está implementada en un motor de bases de datos SQL Server.

Teniendo claro esto, la aplicación móvil será desarrollada en Android junto con Java, por lo cual será necesario importar el paquete de conexión a SQL Server, para establecer la conexión a esta.

La aplicación tendrá un módulo de ingreso general, denomino muchas veces como “Login” el cual establecerá una conexión al Servicio Web el cual realizara la petición a la base de datos, para poder realizar la auto identificación necesaria. Posterior a esto, se manejará una especie de “Session Persistente” la cual estará activa mientras la aplicación esté “ejecutándose”. Continuo a esto, se realiza una segunda consulta SQL, la cual pretende obtener una “Guía Válida” basándose en el RUT del operario de carga, la fecha de parte y el estado valido de la Guía. En caso de que no existan guías asociadas a un operario (algo que puede suceder en festivos, días de descanso, etc.) la aplicación se mantendrá en un modo de “espera”.

El módulo de trazabilidad, se subdivide en un módulo de reporte, el cual consta de una pantalla con área de texto, en la cual el usuario podrá estar escribiendo, enviando mensajes e informando de su estado. Este módulo, genera una ventana de confirmación para cada uno de los reportes que se pretendan enviar. Los mensajes enviados no pueden ser cancelados, y sólo al momento de ser recibidos por el operador de planta, podrán ser modificados. Este módulo implementa el envío de e-mails a través del API interna de Android.

El operario de planta, recibe cada uno de los reportes enviados por los operadores de carga.

Cada mensaje es validado y confirmado para que sea insertado en la base de datos y sea una información válida y esté disponible para el cliente. Este módulo descrito, es el que se conoce como “Modulo de Validación”. Los mensajes enviados desde el “Modulo Reporte” son recibidos en el “Modulo de Validación” en tiempo real, sin existir intermediarios de ningún tipo.

Al terminar de realizar la ruta, el operario de carga finaliza la aplicación, usando el “Modulo de Finalización” el cual sólo se encarga de dejar en “modo espera” a la aplicación.

Cuando la aplicación se encuentra en “modo espera”, está constantemente realizado peticiones simples a la base de datos, a través del mismo “Modulo de Reportes”, en busca de alguna ficha activa para volver a realizar el ciclo de la Trazabilidad, anteriormente descrito.

4.1 Factibilidad Técnica

4.1.1 Tecnología a utilizar para el desarrollo

Para poder implementar la aplicación móvil que se ha analizado en profundidad fue necesario optar por alguna de las tecnologías descritas en el “*Estado del Arte*”.

Debido a la poca masificación de “Windows Phone” fue descartado inmediatamente, debido a que si se implementaba un Web Service para gestionar las peticiones, se podría volver algo complejo realizar una implementación desde esa plataforma de desarrollo.

Se decidió por Android en vez de J2ME, sólo por algunas razones específicas, las cuales se detallan a continuación:

- Android tiene la particularidad de poder instalarse en prácticamente todo **tipo de dispositivo móvil**, lo cual hace que esté presente en los mercados más amplios y tenga un mayor crecimiento. A su vez, es capaz de adaptarse a la perfección a cualquier tipo de necesidad del usuario.
- Al ser “código abierto”, da la posibilidad de poder mejorar su código como también modificarlo. Al existir una gran cantidad de desarrolladores a nivel global, en caso de existir algún tipo de error, este será detectado de manera más rápida y reparado con mayor rigurosidad al no existir ninguna traba legal para poder indagar en su código.
- Android posee una mejor gestión al momento de existir gran cantidad de aplicaciones ejecutándose al mismo tiempo y dejando en estado de “suspensión” aquellas que no se utilicen.

La versión de Android a utilizar es **2.3.3 (API Level 10)**, debido principalmente a su cuota de uso, la cual está graficada en la **Figura 3**.

Por otra parte, sólo se brindó la posibilidad de utilizar el motor de bases de datos “SQL Server 2005” que maneja la empresa, por lo cual está será la utilizada a lo largo de todo el desarrollo, en la cual estarán los datos que se manejan por todo software de la empresa, como también de la aplicación móvil.

Para poder implementar la comunicación entre la aplicación móvil y el servidor, tenemos dos opciones que analizaremos para posteriormente inclinarlos por la que nos ofrezca un desarrollo más ágil.

4.2 Requerimientos Funcionales

ID	Descripción
1	Sistema debe validar el ingreso de los operarios de carga.
2	Sistema es capaz de recibir notificaciones cada cierta cantidad de tiempo.
3	Se debe incluir un módulo en el cual se vea el historial de mensajes que se han enviado, en orden descendente (del más nuevo al más antiguo).
4	El sistema debe validar que existan guías de despacho válidas para la fecha actual, para que no esté realizando notificaciones inexistentes en periodos de fechas inválidos.
5	Se debe poder implementar la funcionalidad “sugerir reportes estándares”
6	Todo reporte que se envía debe tener la opción de ser modificado, como también de ser validado instantáneamente.
7	Los reportes, al ser enviados desde el operario de carga, debe ser validado como: que no se envíe vacío, tenga más de cierta cantidad de caracteres y que no posea garabatos.
8	Las notificaciones sólo son enviadas una vez por periodo de tiempo (3.5 HRS).
9	Para poder utilizar la aplicación móvil, necesariamente se debe iniciar sesión.
10	Los reportes enviados no pueden ser manipulados por el operario de carga, sólo por el operario de planta.
11	La información que es visible para el cliente, se entrega a través de una interfaz anexa a la aplicación móvil. La aplicación sólo se encarga de alimentar de datos a la BD.
12	Cada vez que un operario valide un reporte enviado, se envía un email al cliente, lo cual se maneja de manera externa a la aplicación móvil y su funcionamiento posterior no depende de esta.
13	La aplicación web encargada de insertar los datos a SQL Server cuenta con un sistema de seguridad, encargado de sólo recibir peticiones desde rangos de IP de la red local.
14	Las sugerencias de búsquedas, son previamente agregadas a una tabla específica en la BD, y bajo ningún caso las generadas por los operarios de carga pasarán a formar parte de estas.
15	En caso de existir una caída de la BD, los reportes enviados serán guardados en la memoria local del dispositivo móvil, para que al momento de la reactivación del servicio, sean enviados en el orden en el cual fueron almacenados. (Android utiliza SQLite, el cual utilizaremos para este caso).
16	La aplicación móvil no permite ni incorpora un módulo de registro, por un asunto de regla de negocio, todos estos datos se extraen directamente desde la BD de la empresa, y son gestionados por una aplicación externa.
17	En caso de que la comunicación entre las aplicaciones falle, se almacenan los reportes en el dispositivo móvil, de la misma manera que en el punto 15.

Tabla 5, Requerimientos Funcionales

4.3 Requerimientos No Funcionales

Agruparemos estos requerimientos en 5 tipos:

- **Escalabilidad:** El diseño debe contemplar el uso óptimo de los recursos, como la conexión a la base de datos, el uso de la batería, etc. Para asegurar la escalabilidad el sistema debe reaccionar y adaptarse a posibles caídas del Servicio Web, como también a la conexión a Internet sin perder la información que se está intentando enviar. También se debe contemplar la partición entre los datos, recursos y aplicaciones para optimizar la escalabilidad del sistema.
- **Disponibilidad:** El sistema debe tener una disponibilidad continua, con un servicio para los usuarios de 7 días a la semana x 24 horas, garantizando un esquema de contingencia que permita una solución frente a una posible falla, como también, ofrecer consistencia al producirse una recuperación del sistema, para que no exista pérdida de información y por sobre todo que contemplen la interrupción de transacciones para que éstas finalicen de forma correcta.
- **Seguridad:** La solución propuesta, debe reflejar ciertos patrones de seguridad, teniendo en consideración la alta sensibilidad de la información que se va a manejar. Al ser una aplicación de tipo “privada” no es accesible por cualquier persona, por lo cual sólo se podrá utilizar para ciertos equipos, utilizando algún “llave de acceso”.
- **Mantenibilidad:** El sistema debe ser construido de tal manera que en caso de existir algún cambio en los parámetros de negocio no obligue a la generación de una nueva versión del módulo. Se debe considerar los cambios frecuentes que se realizan en el esquema de la base de datos, para lo cual se debe contemplar la obtención de la información basándose en la meta-data de las tablas de la base de datos, para que a pesar de los cambios realizados, sea posible continuar con el normal funcionamiento de la aplicación.
- **Desempeño:** El diseño debe contemplar un buen desempeño ante una alta demanda, siendo capaz de tener un buen tiempo de respuesta al momento de realizar algún tipo de operación, ya sea ingreso de un usuario como también el envío de información. Debe considerar la alta concurrencia a la información en el servidor de datos que se produce en las horas de alta concurrencia de la empresa, ofreciendo un sistema que pueda obtener la información a pesar de existir una gran demanda por parte de los trabajadores internos, externos y los usuarios finales de la aplicación móvil.

4.4 Casos de Uso

4.4.1 Caso de Uso de Alto Nivel

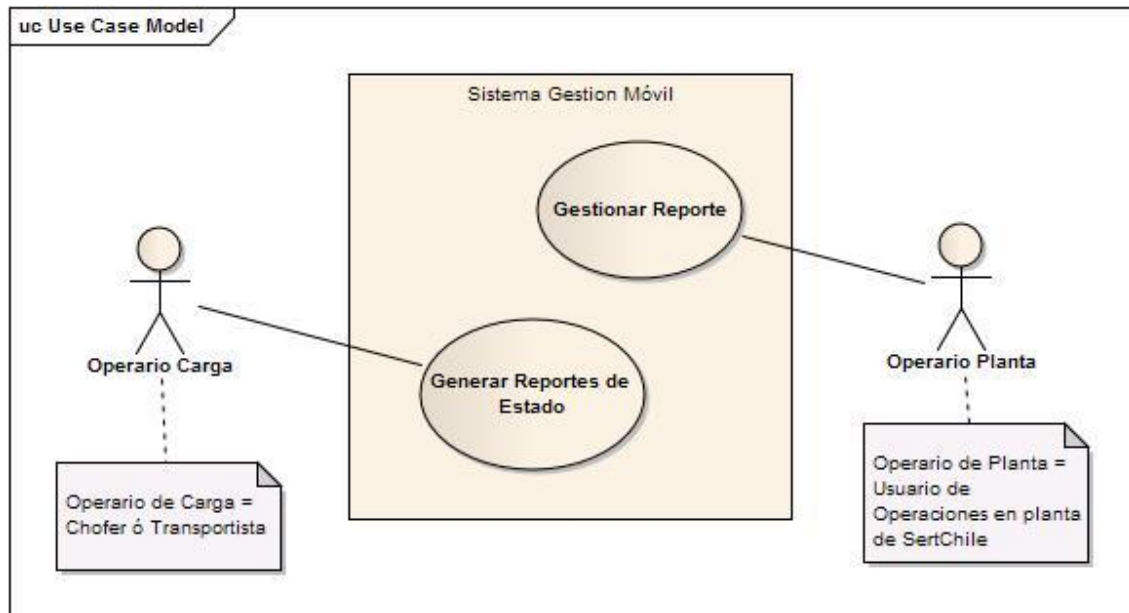


Figura 8, Caso de Uso de Alto Nivel

4.4.2 C.U: Generar Reportes de Estado

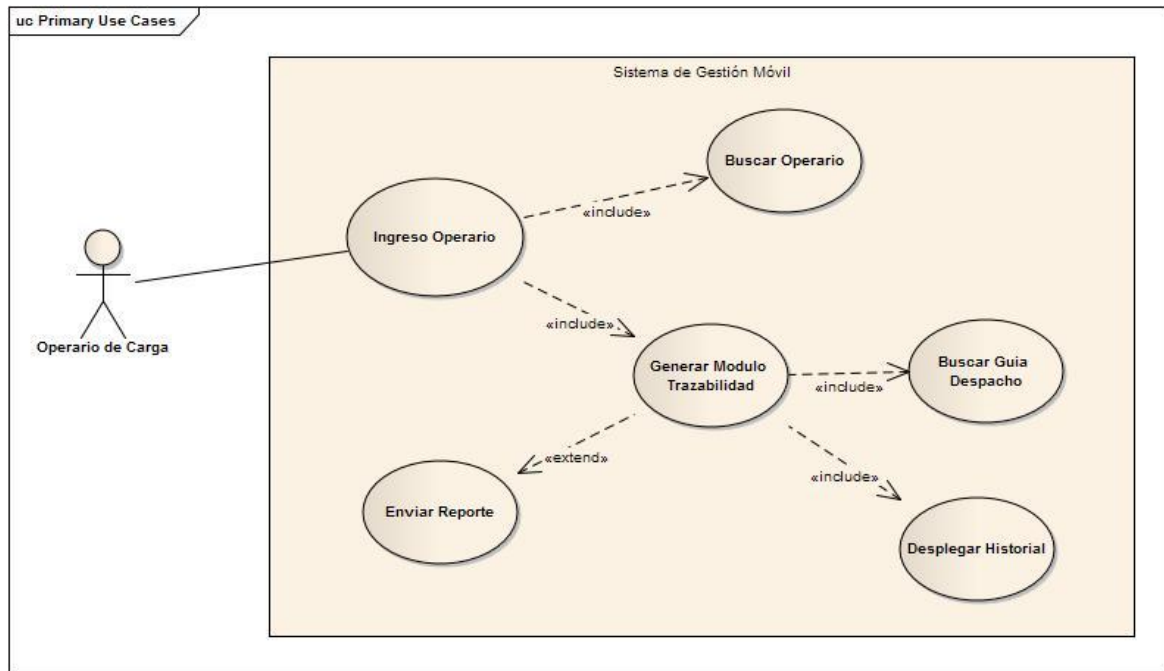


Figura 9, Caso de Uso Generar Reportes de Estado

4.4.2.1 C.U 1.1: Ingreso Operario

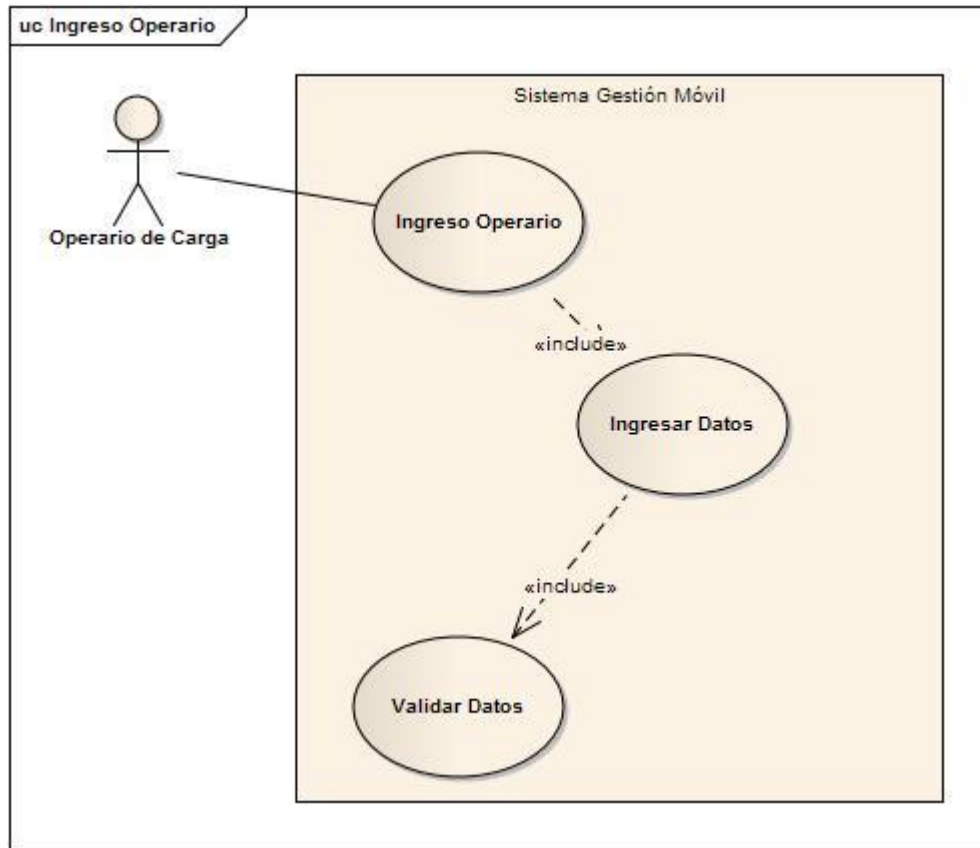


Figura 10, Caso de Uso Ingreso Operario

Tabla 6, Caso de Uso Ingresar datos

ID	CU 1.1.1
Caso de Uso	Ingresar datos
Actores	Operario de carga
Tipo	Primario
Descripción	Se realiza el ingreso del operario, ingresando sus datos para realizar esta operación, los cuales son RUT y su contraseña

Tabla 7, Caso de Uso Validar Datos

ID	CU 1.1.2
Caso de Uso	Validar Datos
Actores	Operario de carga
Tipo	Primario
Descripción	Se realiza el ingreso del operario, ingresando sus datos para realizar esta operación, los cuales son RUT y su contraseña

4.4.2.2 C.U 1.2: Buscar Operario

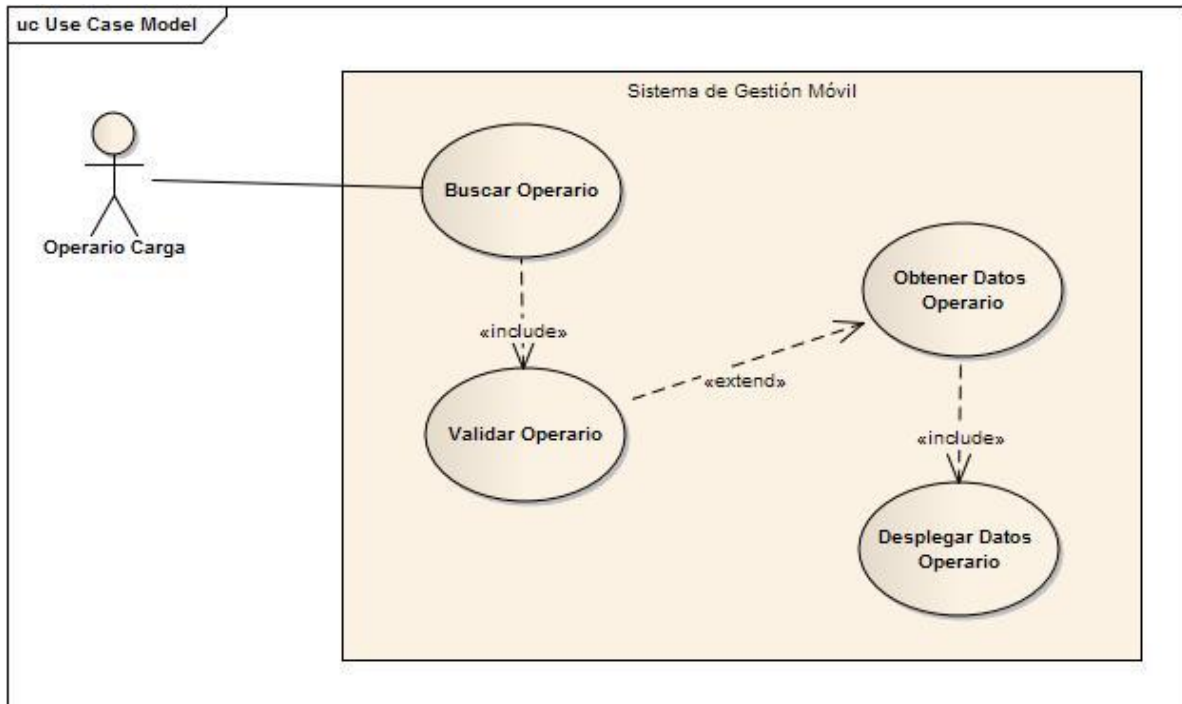


Figura 11, Caso de Uso Buscar Operario

Tabla 8, Validar Operario

ID	CU 1.2.1
Caso de Uso	Validar Operario
Actores	Operario de carga
Tipo	Primario
Descripción	Se valida que el RUT del operario exista como también que se encuentre registrado.

Tabla 9, Caso de Uso Obtener datos Operario

ID	CU 1.2.2
Caso de Uso	Obtener datos Operario
Actores	Operario de carga
Tipo	Primario
Descripción	Se obtienen los datos asociados al operario, como su nombre, teléfono y camión asignado

Tabla 10, Caso de Uso Desplegar Datos Operario

ID	CU 1.2.3
Caso de Uso	Desplegar Datos Operario
Actores	Operario de carga
Tipo	Primario
Descripción	Se muestran los datos obtenidos en el CU 1.2.3

4.4.2.3 C.U 1.3: Generar Modulo de Trazabilidad

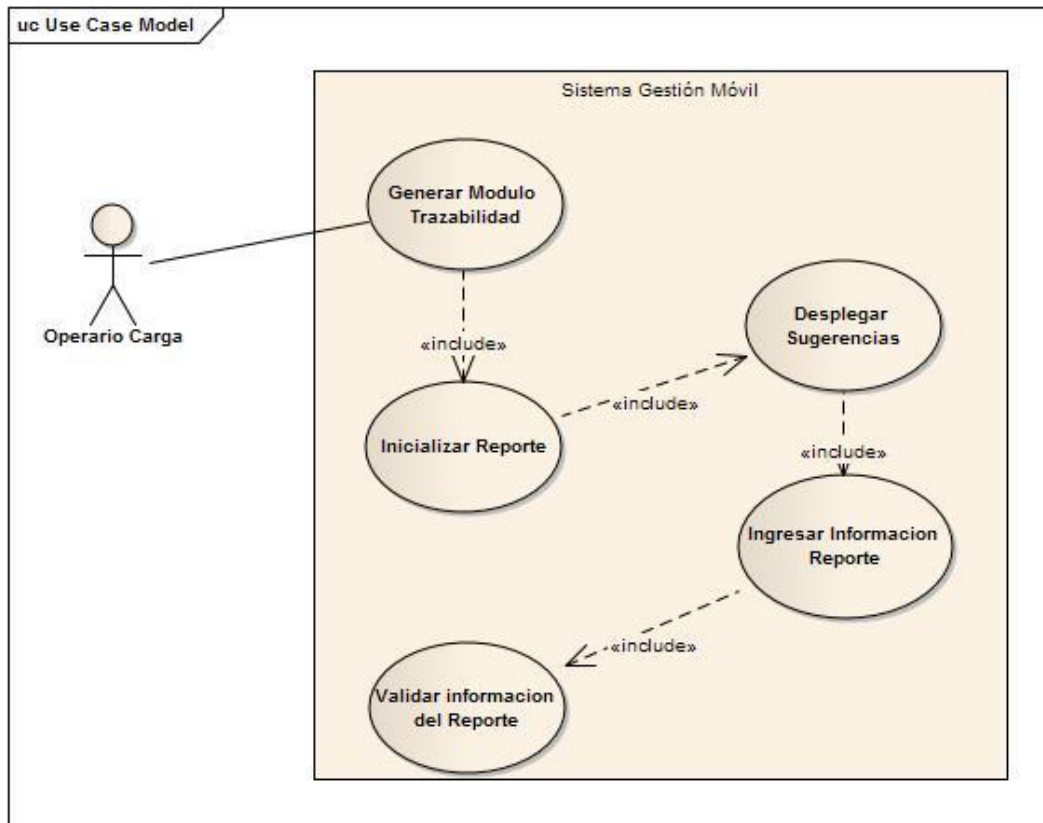


Figura 12, Caso de Uso Generar Modulo de Trazabilidad

Tabla 11, Caso de Uso Generar Modulo de Trazabilidad

ID	CU 1.3.1
Caso de Uso	Inicializar Reporte
Actores	Operario de carga
Tipo	Primario
Descripción	Se activa la opción de generar reportes

Tabla 12, Caso de Uso Desplegar Sugerencias

ID	CU 1.3.2
Caso de Uso	Desplegar Sugerencias
Actores	Operario de carga
Tipo	Primario
Descripción	Despliega ciertos reportes estándar para que sean ingresados automáticamente

Tabla 13, Caso de Uso Ingresar Información Reporte

ID	CU 1.3.3
Caso de Uso	Ingresar Información Reporte
Actores	Operario de carga
Tipo	Primario
Descripción	Se agregan comentarios del estado actual de lo que está realizando el operario de carga

Tabla 14, Caso de Uso Validar Información del Reporte

ID	CU 1.3.4
Caso de Uso	Validar Información del Reporte
Actores	Operario de carga
Tipo	Primario
Descripción	A nivel de aplicación móvil se valida que el mensaje no este vacío, que tenga más de una cantidad específica de caracteres, etc.

4.4.2.4 C.U 1.4: Buscar Guía Despacho

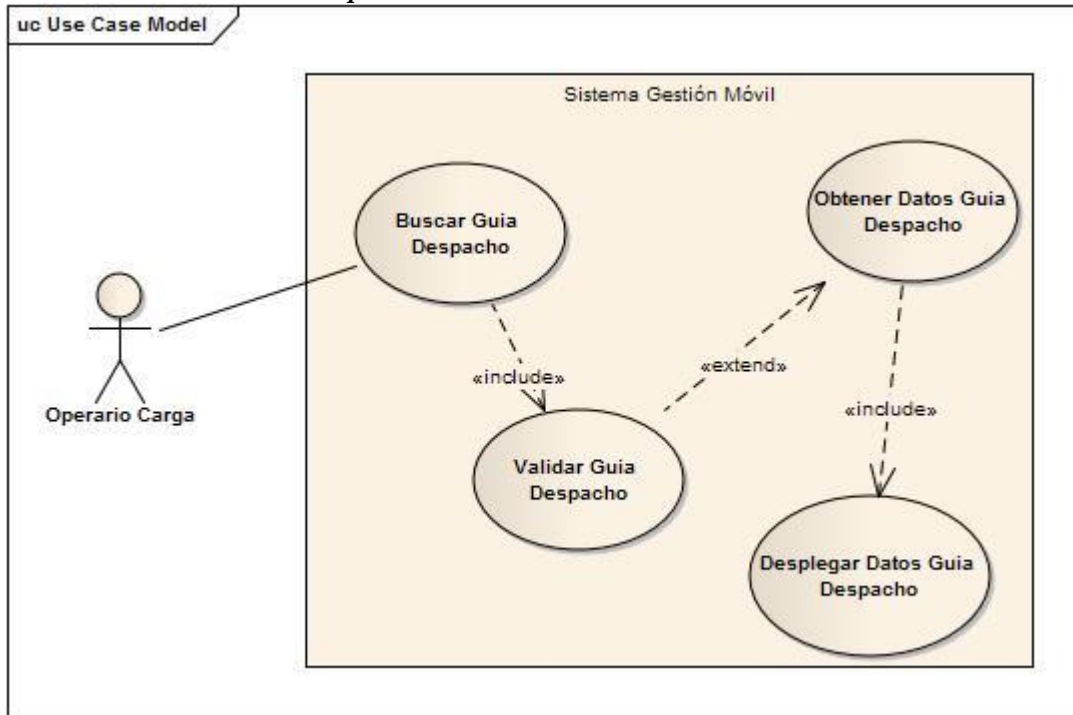


Figura 13, Caso de Uso Buscar Guía Despacho

Tabla 15, Caso de Uso Buscar Guía Despacho

ID	CU 1.4.1
Caso de Uso	Validar Guía Despacho
Actores	Operario de carga
Tipo	Primario
Descripción	Se valida si existe una guía para la fecha de la consulta, en donde el estado de esta sea “activo”.

Tabla 16, Caso de Uso Buscar Guía Despacho

ID	CU 1.4.2
Caso de Uso	Obtener datos Guía Despacho
Actores	Operario de carga
Tipo	Primario
Descripción	Se obtienen los datos para la guía, tales como nombre del cliente, carga a transportar, fecha de partida, etc.

Tabla 17, Caso de Uso Desplegar Datos Guía Despacho

ID	CU 1.4.3
Caso de Uso	Desplegar Datos Guía Despacho
Actores	Operario de carga
Tipo	Primario
Descripción	De los datos obtenidos, se procede a desplegar información asociada a la carga.

4.4.2.5 C.U 1.5: Enviar Reporte

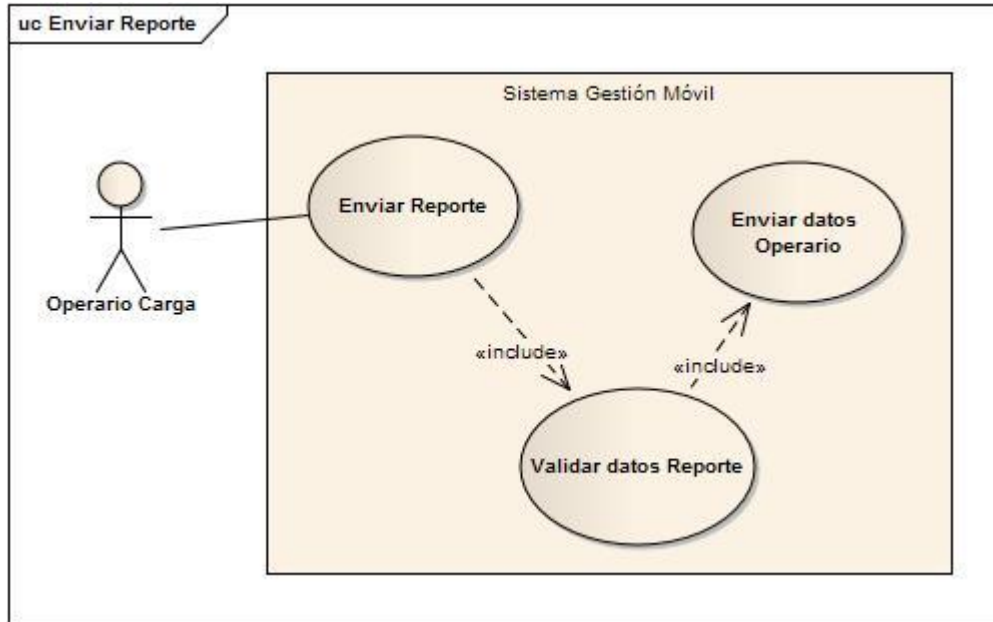


Figura 14, Caso de Uso Enviar Reporte

Tabla 18, Caso de Uso Validar Datos del Reporte

ID	CU 1.5.1
Caso de Uso	Validar Datos del Reporte
Actores	Operario de carga
Tipo	Primario
Descripción	Se valida que el reporte que se está tratando de enviar no este vacío y que la cantidad de caracteres sea mayor a 10 caracteres.

Tabla 19, Caso de Uso Enviar Datos al Operario

ID	CU 1.5.2
Caso de Uso	Enviar Datos al Operario
Actores	Operario de carga, Operario de Planta
Tipo	Primario
Descripción	Se envía por un e-mail al operario de planta el reporte generado.

4.4.2.6 C.U 1.6: Desplegar Historial

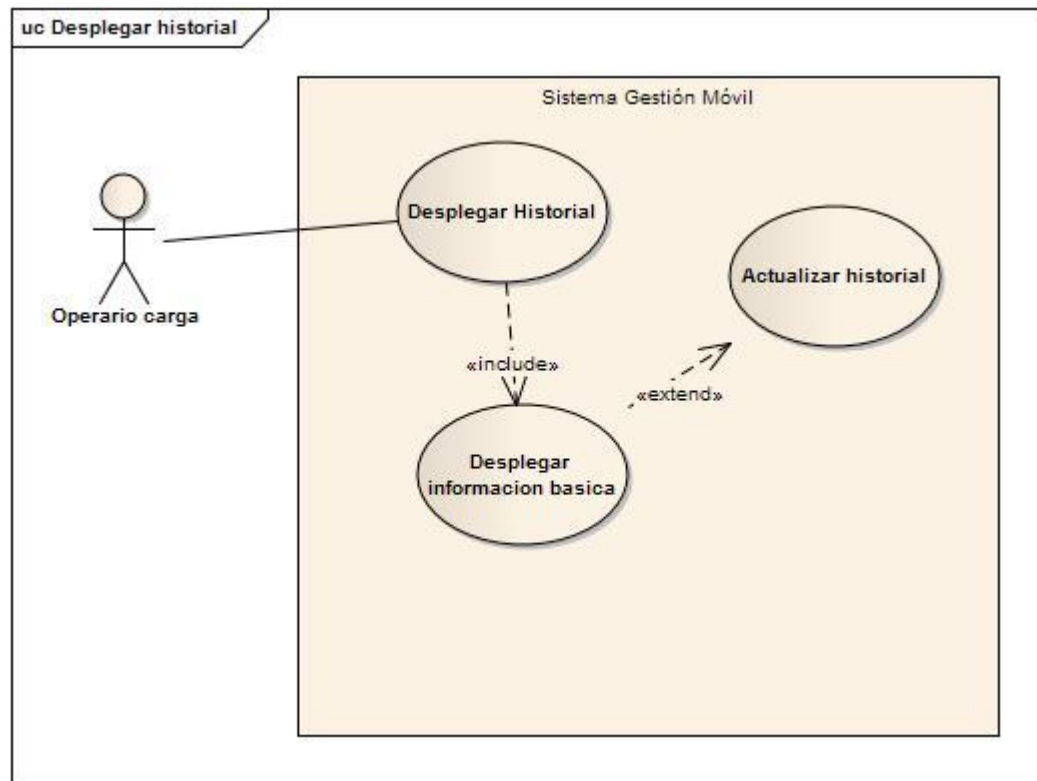


Figura 15, Caso de Uso Desplegar Historial

Tabla 20, Caso de Uso Desplegar Información Básica

ID	CU 1.6.1
Caso de Uso	Desplegar Información Básica
Actores	Operario de carga
Tipo	Primario
Descripción	Se despliega el historial básico de reportes enviados desde que fue generado el módulo de trazabilidad.

Tabla 21, Caso de Uso Actualizar Historial

ID	CU 1.6.1
Caso de Uso	Actualizar Historial
Actores	Operario de carga
Tipo	Primario
Descripción	Si existen nuevos reportes enviados, se actualizan los datos del historial actual

4.4.3 C.U: Gestionar Reportes

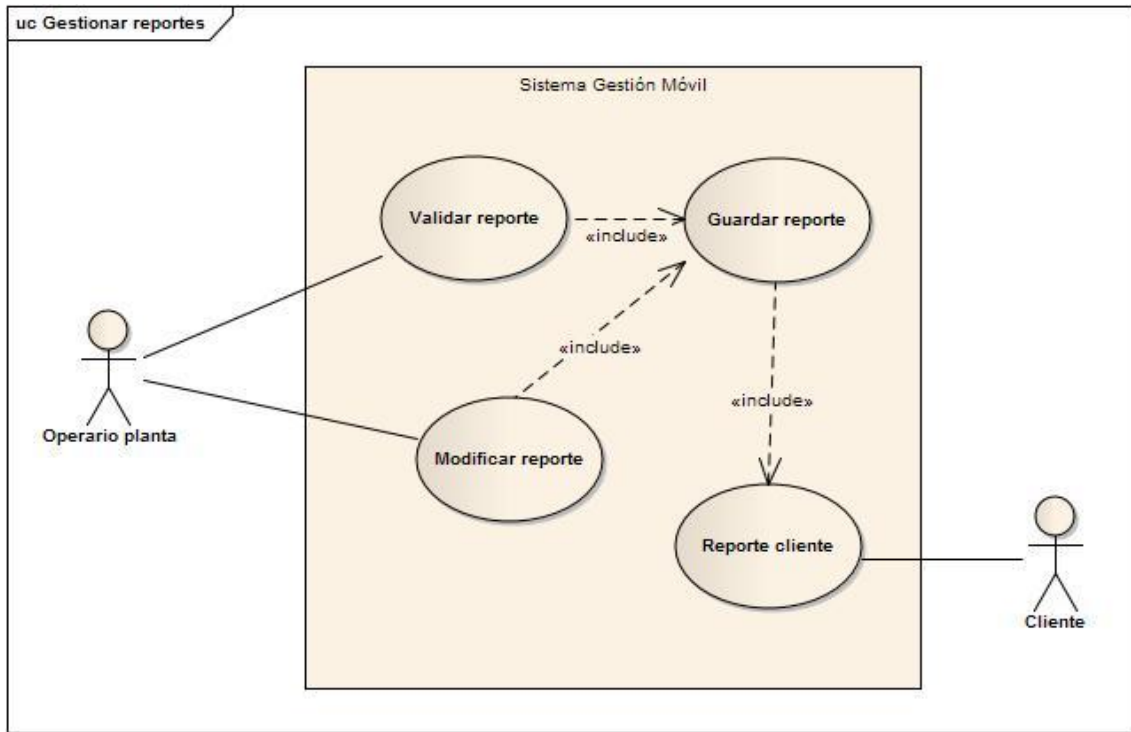


Figura 16, Caso de Uso Gestionar Reportes

4.4.3.1 C.U 2.1: Validar Reporte

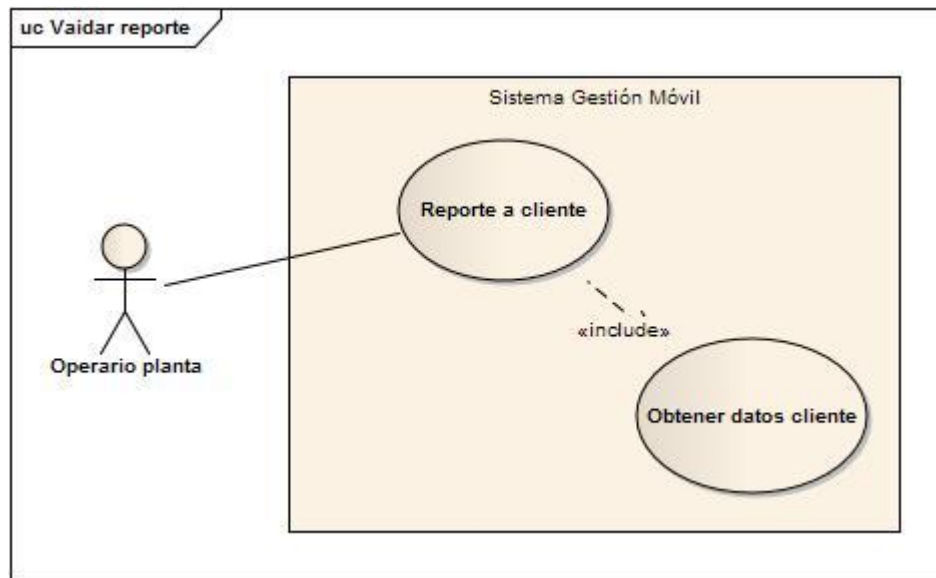


Figura 17, Caso de Uso Validar Reporte

Tabla 22, Caso de Uso Generar Estructura Reporte

ID	CU 2.1.1
Caso de Uso	Generar Estructura Reporte
Actores	Operario de planta
Tipo	Primario
Descripción	Se genera la estructura del archivo a enviar al cliente, considerando información sobre el reporte que ha llegado, la información de la carga e información de carácter interno de la empresa.

4.4.3.2 C.U 2.2: Modificar Reporte

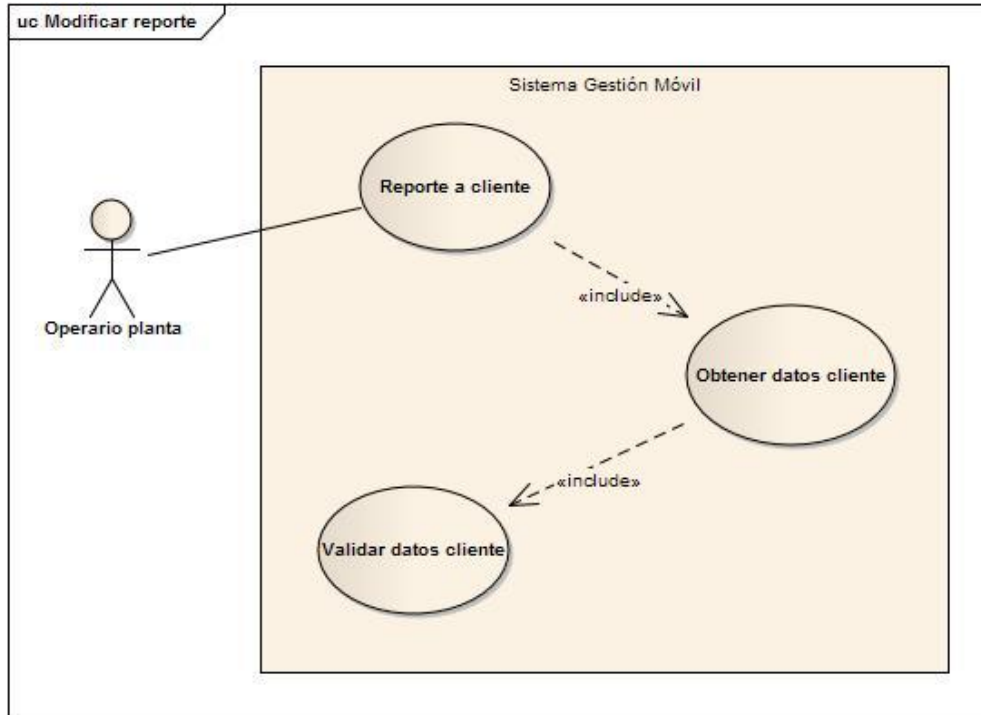


Figura 18, Caso de Uso Modificar Reporte

Tabla 23, Caso de Uso Ingresar Información nueva

ID	CU 2.2.1
Caso de Uso	Ingresar Información nueva
Actores	Operario de planta
Tipo	Primario
Descripción	Se agrega información al reporte actual, considerando ciertas situaciones, como atrasos, congestión en la ruta, cortes de tránsito, etc. Por lo general, es información que recopila el operario de planta que sirve para profundizar y expandir el reporte que va a llegar al cliente.

Tabla 24, Caso de Uso Generar Estructura del Reporte

ID	CU 2.2.2
Caso de Uso	Generar Estructura del Reporte
Actores	Operario de planta
Tipo	Primario
Descripción	Se genera la estructura del archivo a enviar al cliente, considerando información sobre el reporte que ha llegado, la información de la carga e información de carácter interno de la empresa.

4.4.3.3 C.U 2.3: Guardar Reporte

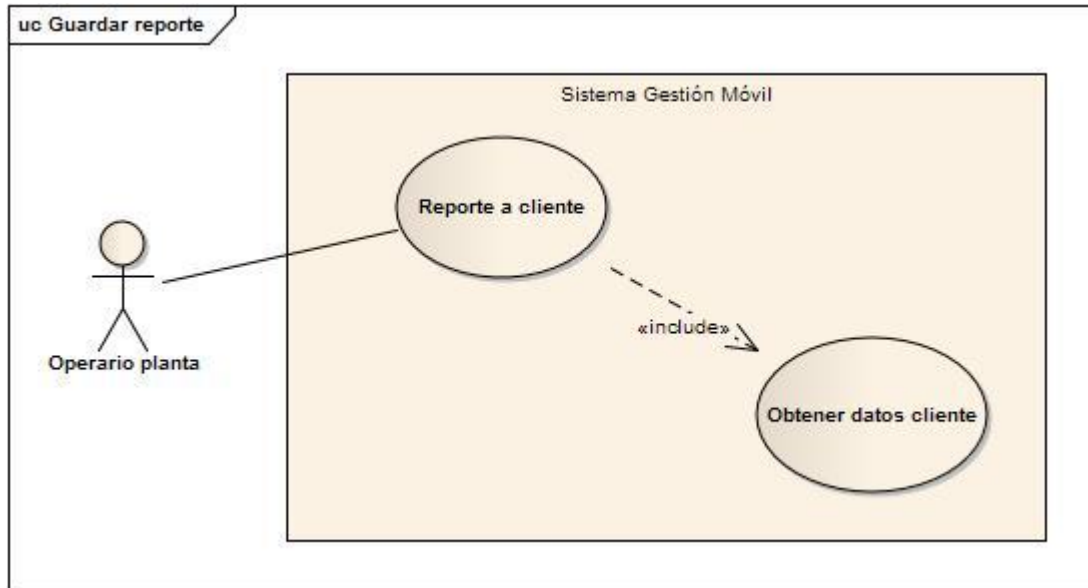


Figura 19, Caso de Uso Guardar Reporte

Tabla 25, Caso de Uso Actualizar Historial Reportes

ID	CU 2.3.1
Caso de Uso	Actualizar Historial Reportes
Actores	Operario de planta
Tipo	Primario
Descripción	Se actualiza la información de historial en un sistema externo.

4.4.3.4 C.U 2.4: Reporte al Cliente

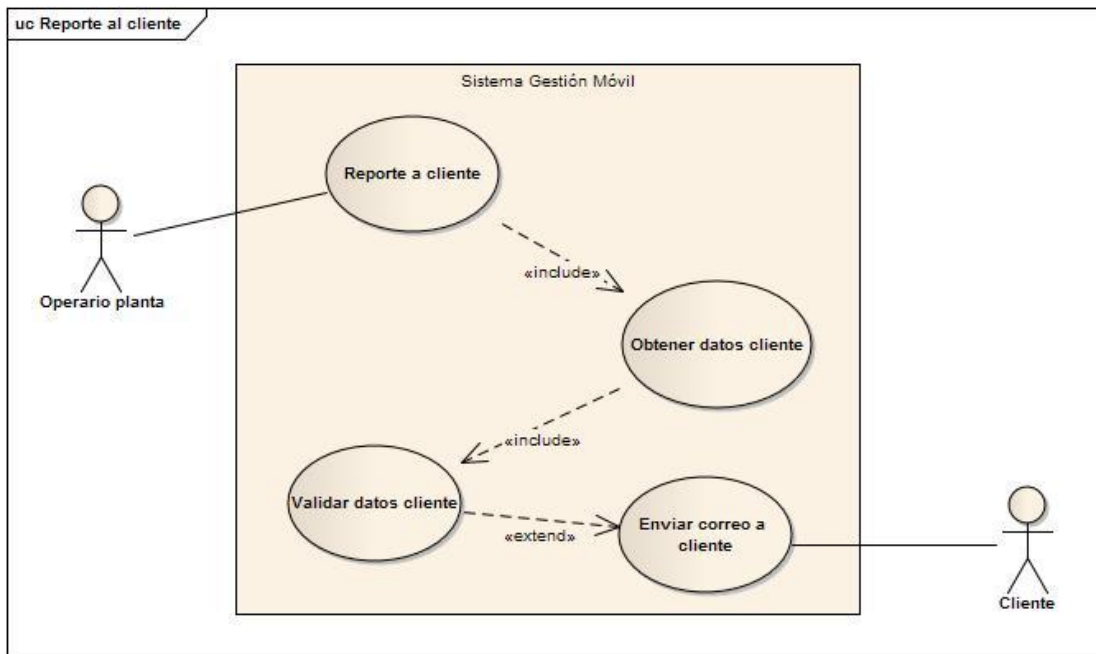


Figura 20, Caso de Uso Reporte Cliente

Tabla 26, Caso de Uso Obtener datos Cliente

ID	CU 2.4.1
Caso de Uso	Obtener datos Cliente
Actores	Operario de planta
Tipo	Primario
Descripción	Se obtienen los datos asociados al cliente para poder generar la estructura de la información que será enviada

Tabla 27, Caso de Uso Validar Datos Cliente

ID	CU 2.4.2
Caso de Uso	Validar Datos
Actores	Operario de planta
Tipo	Primario
Descripción	Se valida que exista un email para el cliente.

Tabla 28, Caso de Uso Enviar Reporte Email

ID	CU 2.4.3
Caso de Uso	Enviar Reporte Email
Actores	Operario de planta, Cliente
Tipo	Primario
Descripción	De existir un email asociado al cliente, se envía el reporte a su casilla de correo.

4.5 Diagramas de Actividad

4.5.1 Actividad: Ingreso Operario

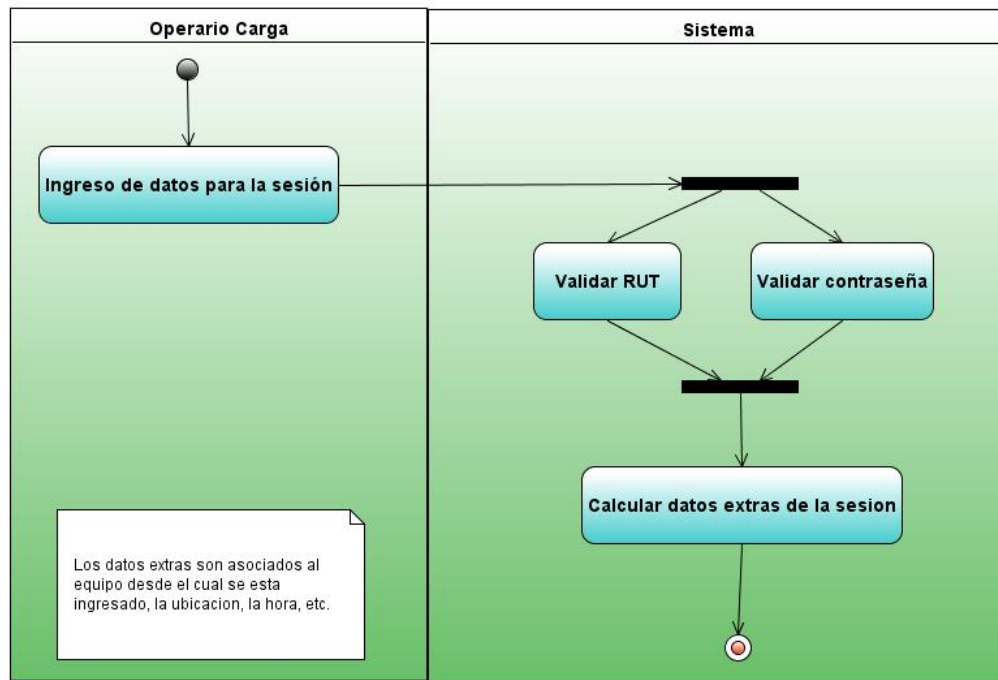


Figura 21, Actividad Ingreso Operario

4.5.2 Actividad: Buscar Operario

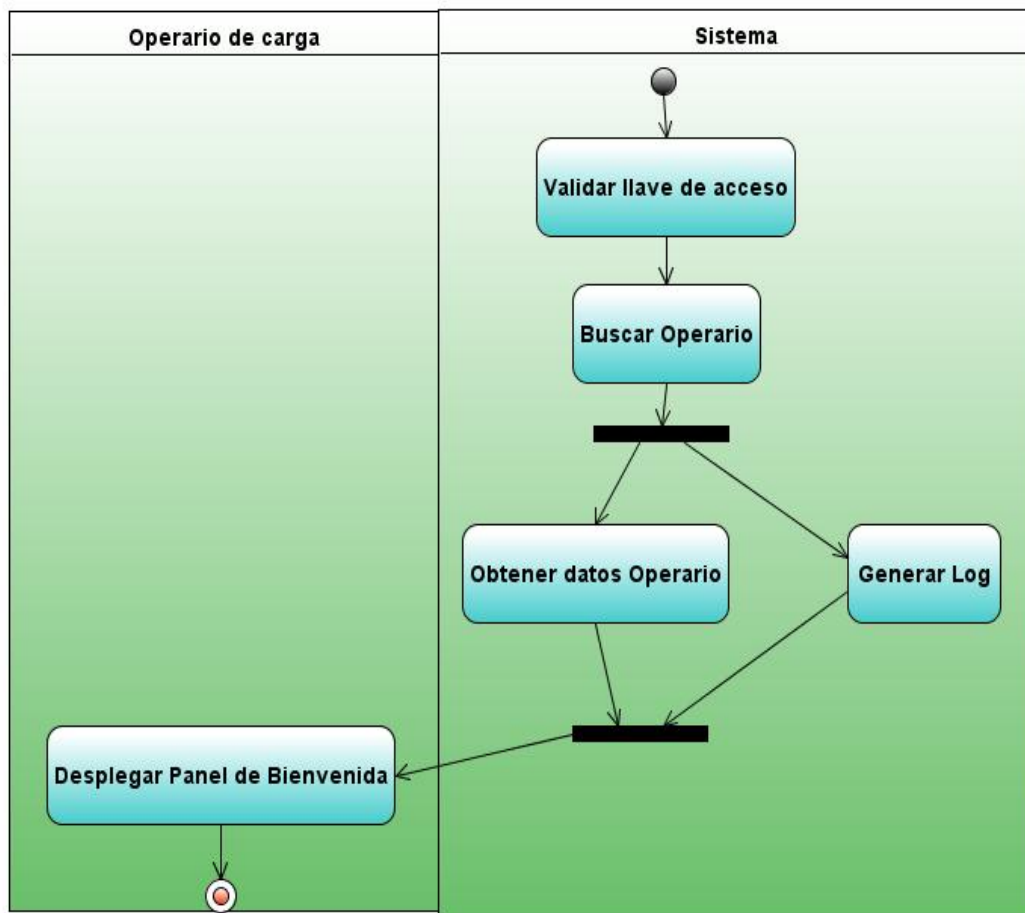


Figura 22, Actividad Buscar Operario

4.5.3 Actividad: Generar Modulo de Trazabilidad

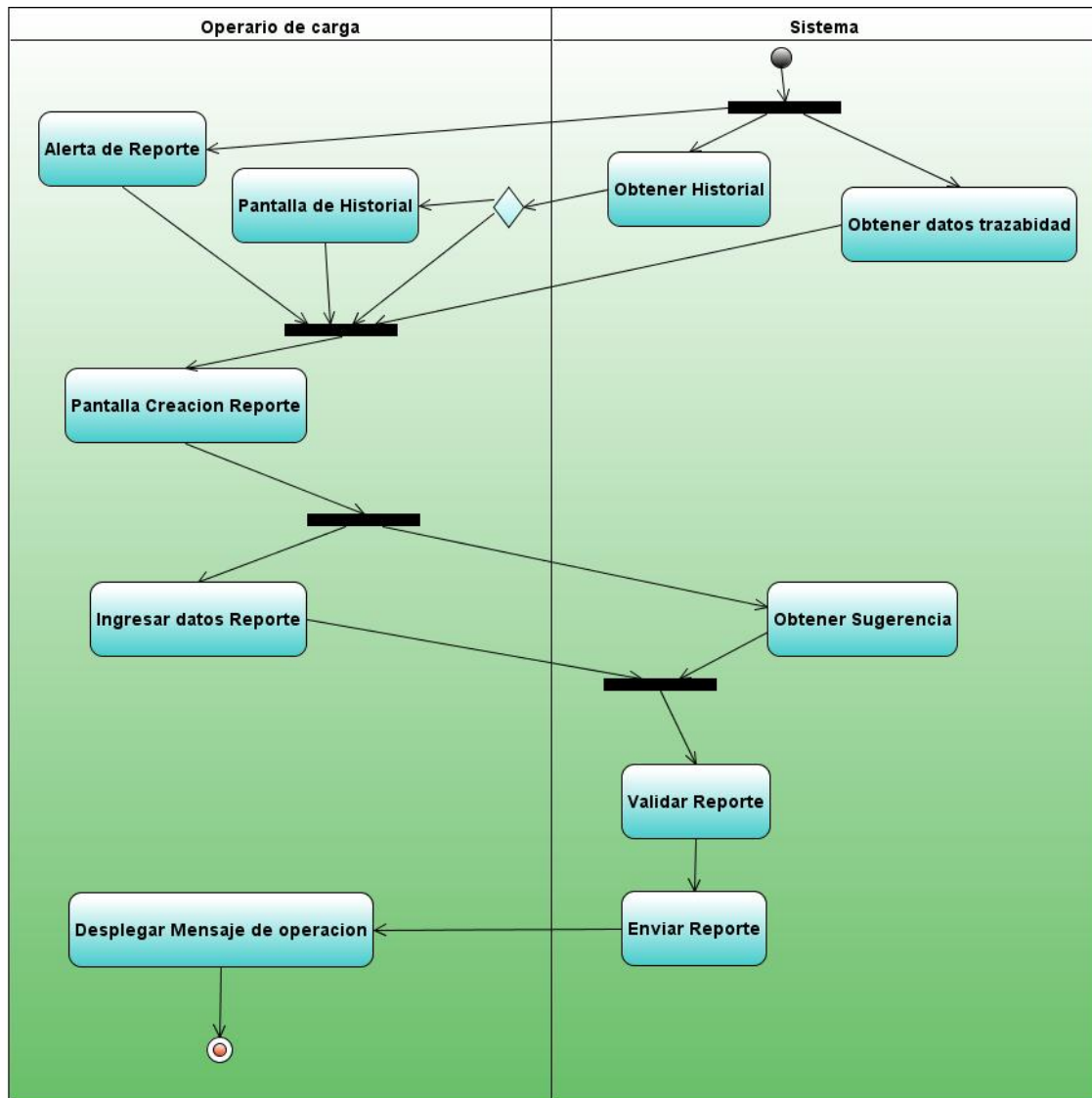


Figura 23, Actividad Generar Módulo Trazabilidad

4.5.4 Actividad: Buscar Guía Despacho

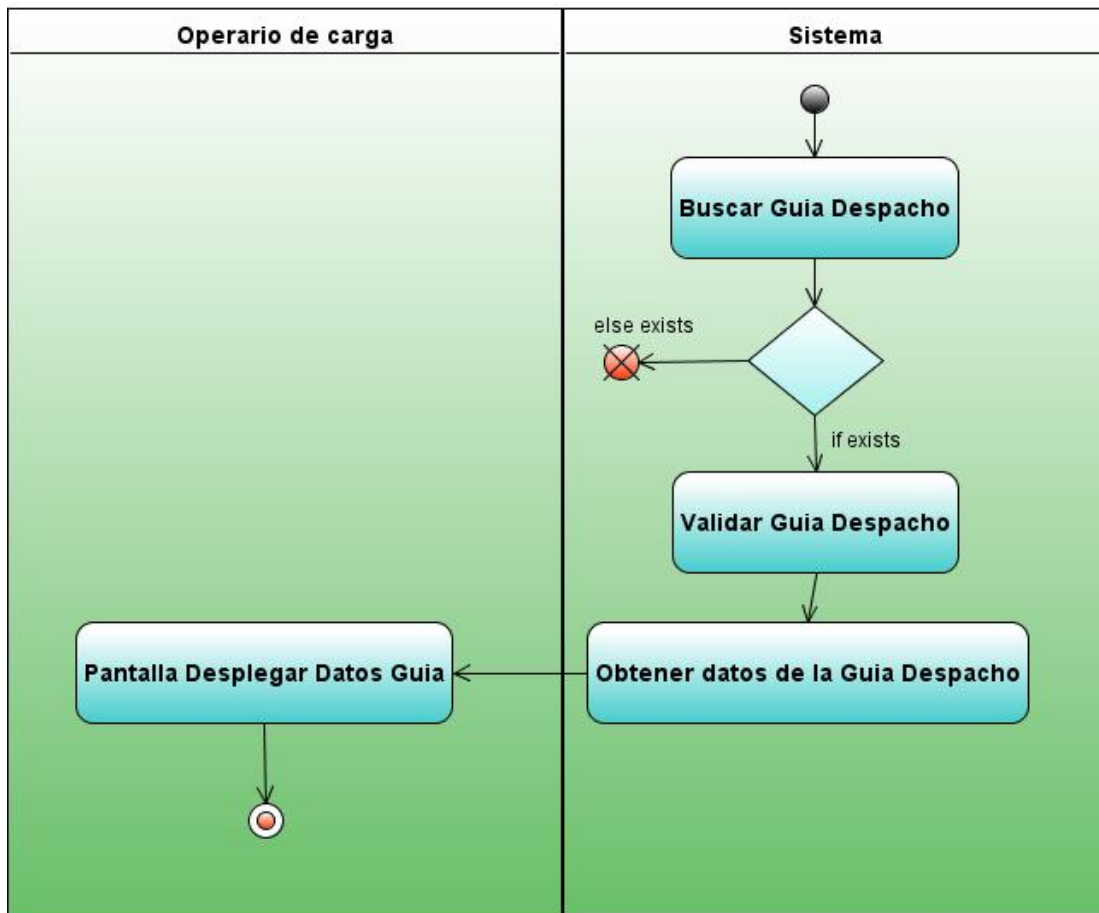


Figura 24, Actividad Buscar Guía Despacho

4.5.5 Actividad: Enviar Reporte

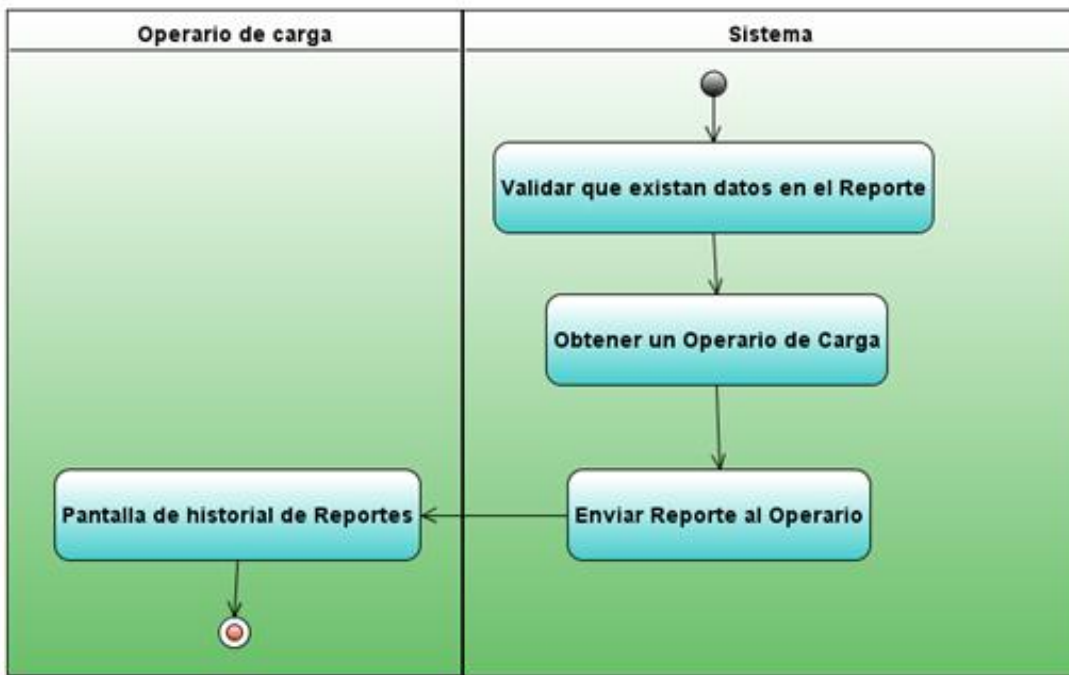


Figura 25, Actividad Enviar Reporte

4.5.6 Actividad: Desplegar Historial

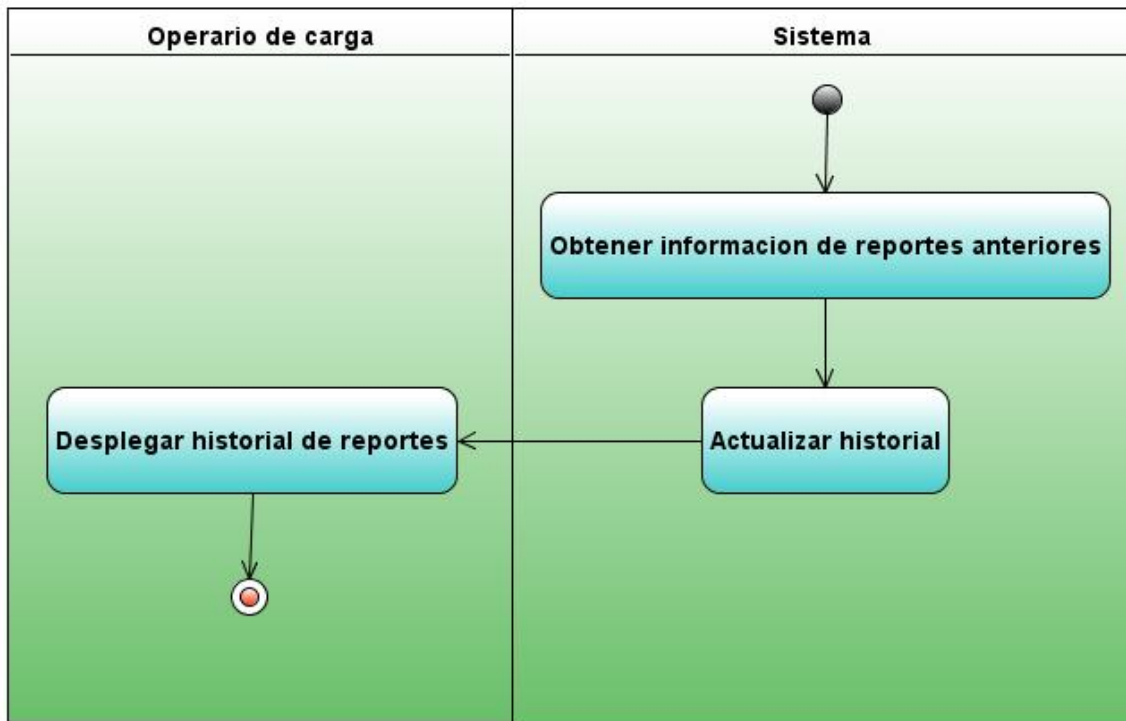


Figura 26, Actividad Desplegar Historial

4.5.7 Actividad: Validar Reporte

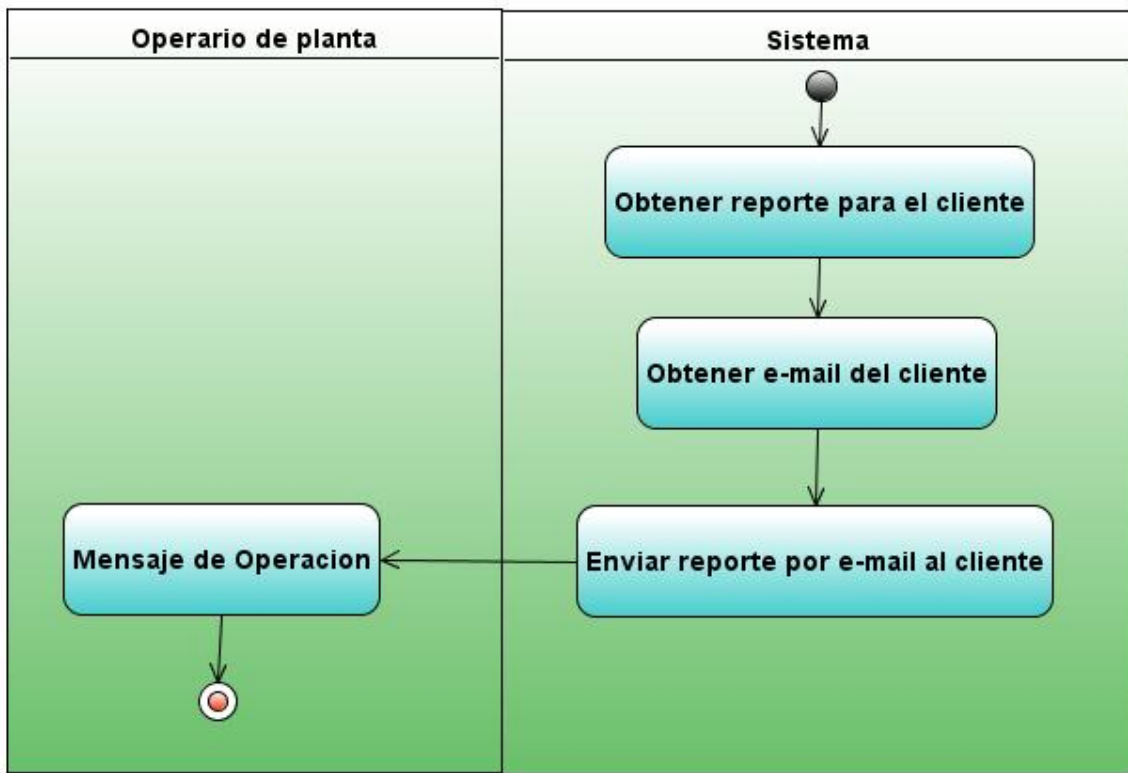


Figura 27, Actividad Validar Reporte

4.5.8 Actividad: Modificar Reporte

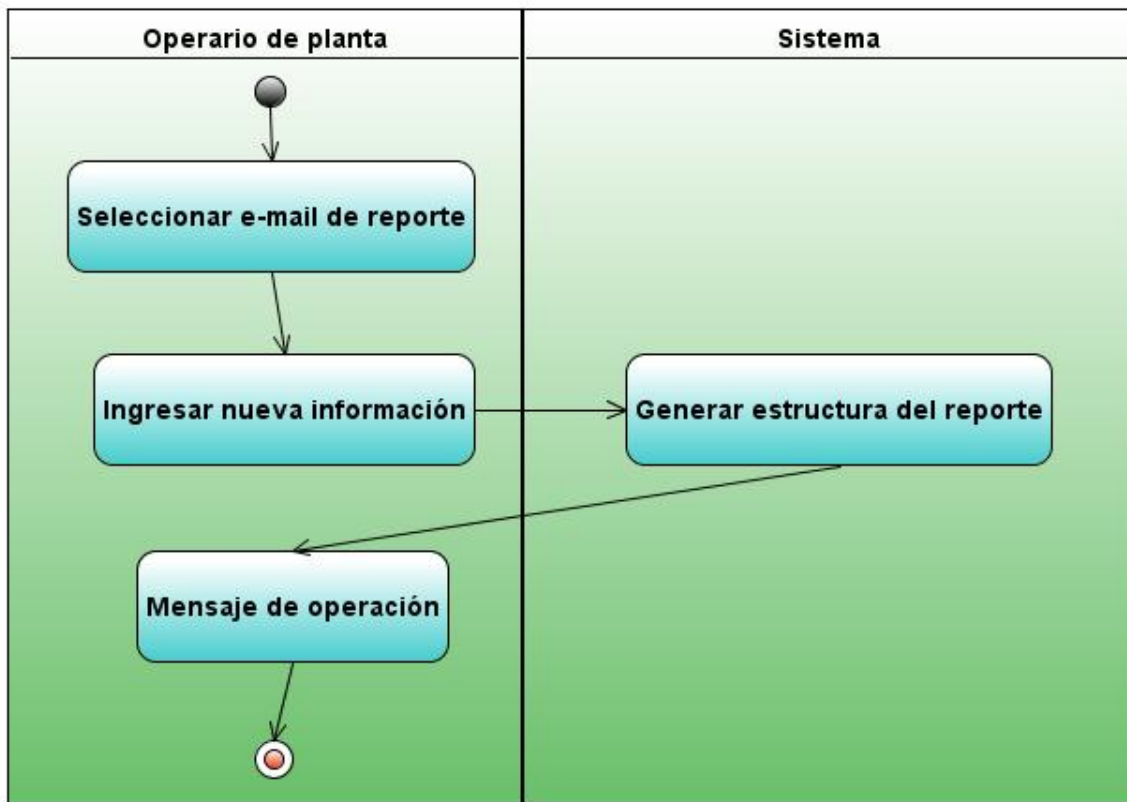


Figura 28, Actividad Modificar Reporte

4.6 Diagramas de Secuencia

4.6.1 Ingreso de operario carga

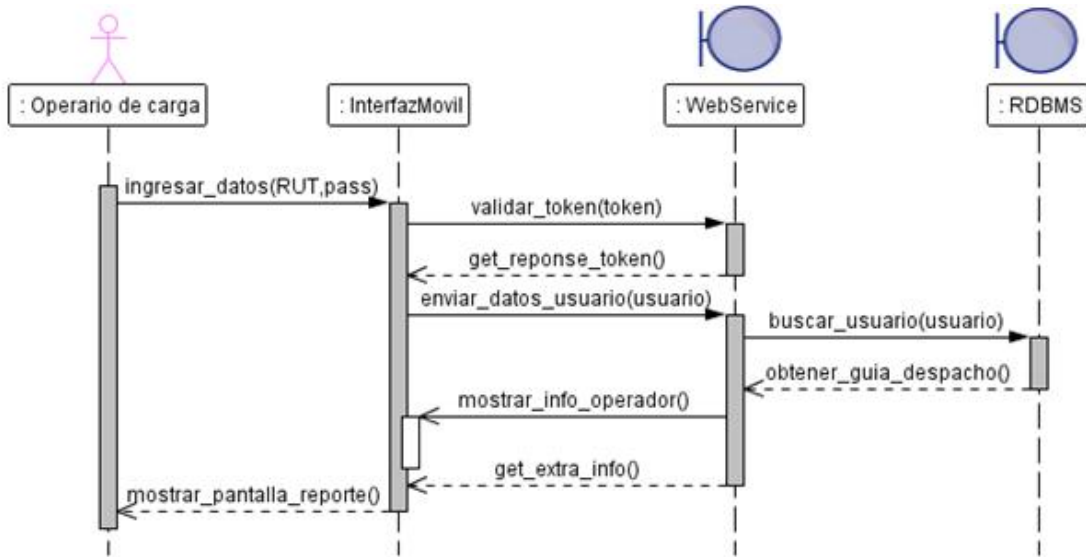


Figura 29, Secuencia Ingreso operario carga

4.6.2 Generación de reporte

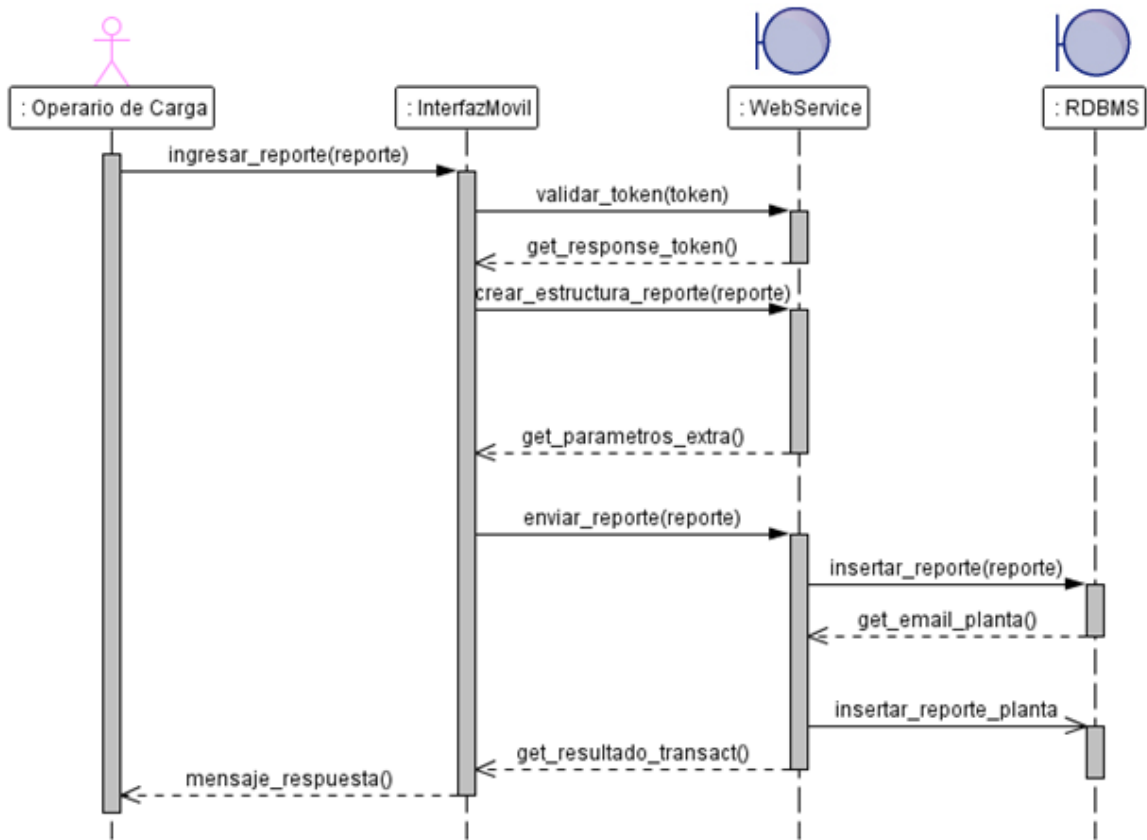


Figura 30, Secuencia Generación de Reporte

4.6.3 Despliegue de Historial

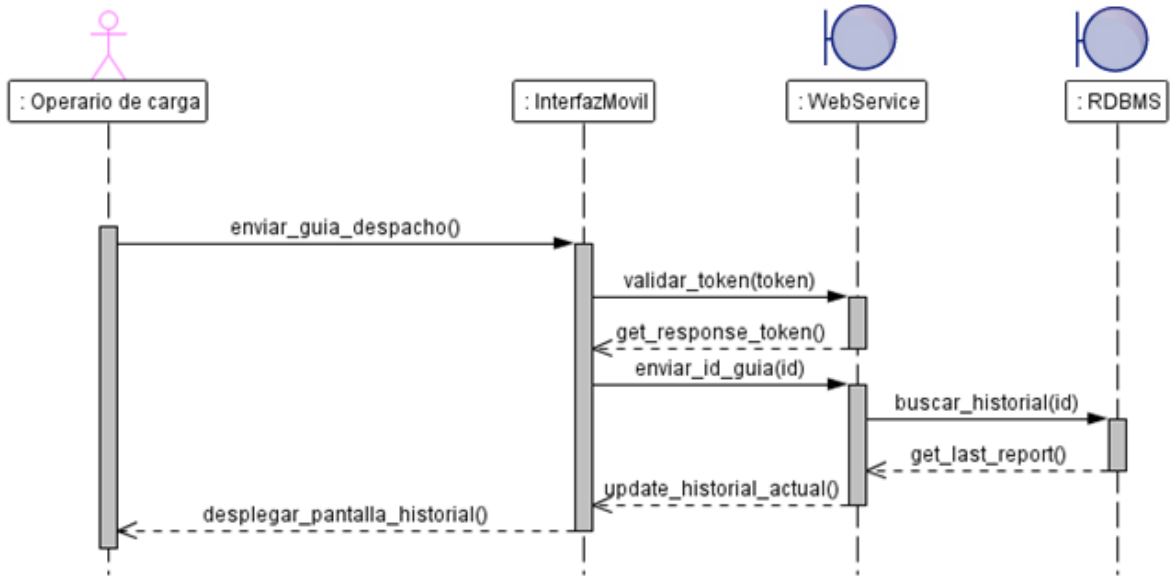


Figura 31, Secuencia Despliegue Historial

4.6.4 Validar/Modificar Reporte

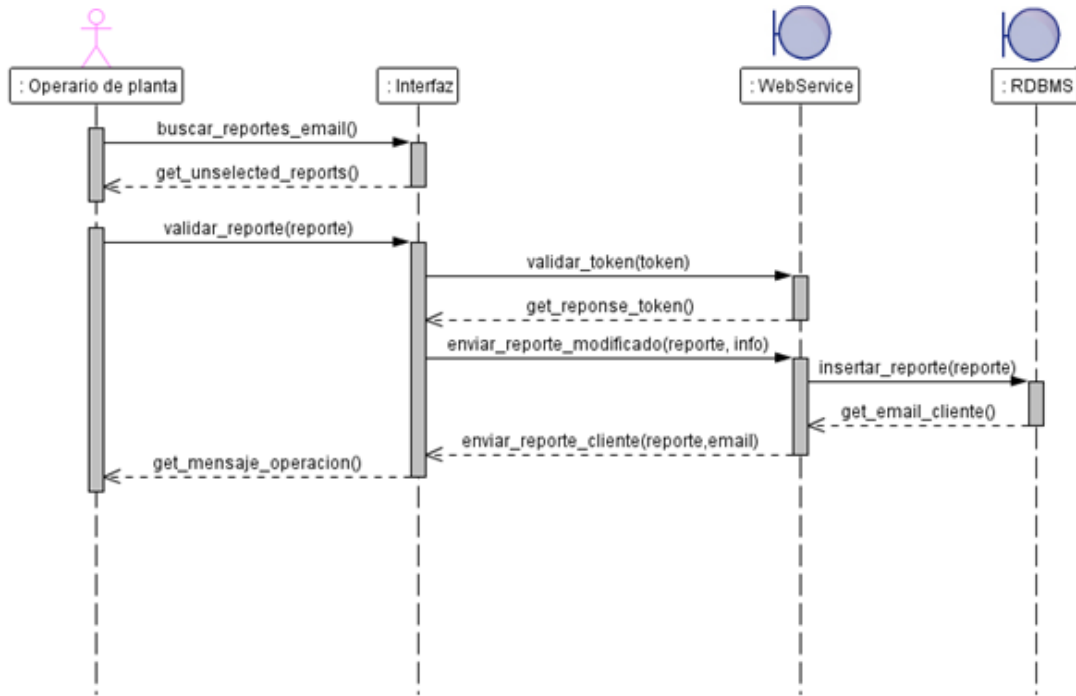


Figura 32, Secuencia Validar/Modificar Reporte

4.7 Diagrama de Clases

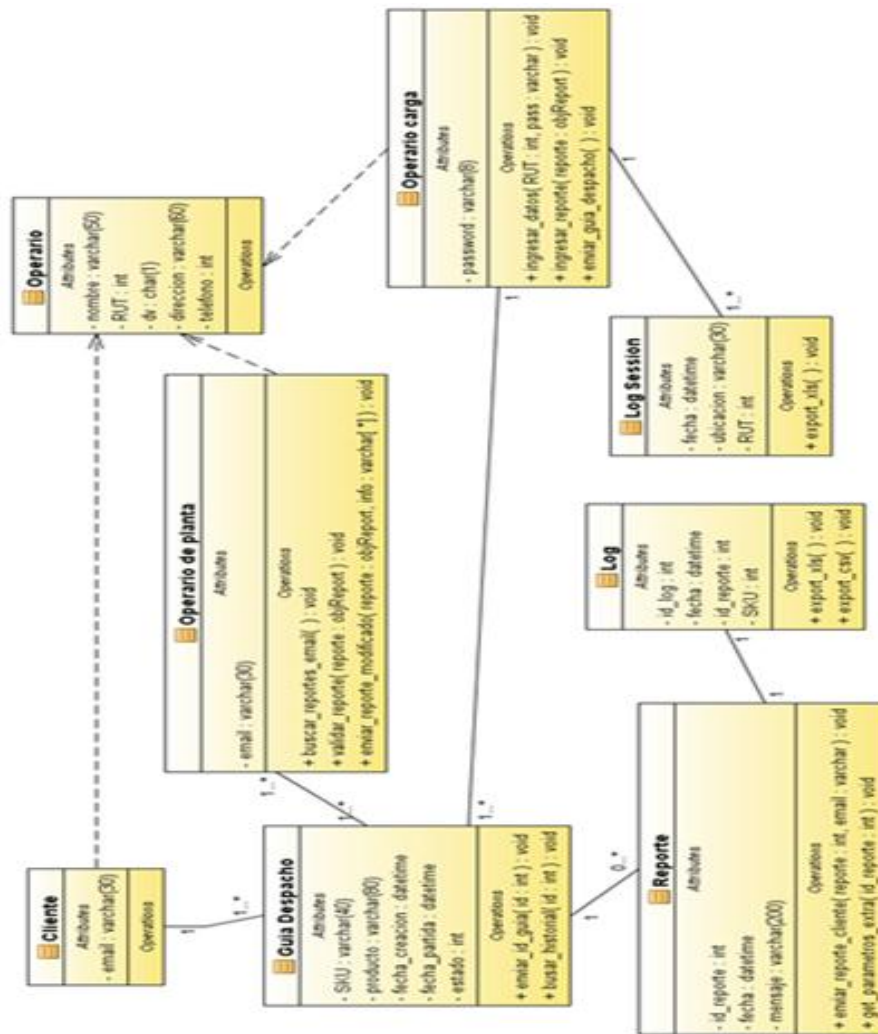


Figura 33, Diagrama de Clases

4.8 Protocolos

4.8.1 Protocolo a utilizar: Protocolo TCP

Técnicamente, se utiliza el protocolo TCP para que las aplicaciones puedan comunicarse de manera segura independiente de cualquier capa inferior.

Durante la comunicación, las dos máquinas establecen una conexión, un denominado cliente y el otro servidor. Ambos, utilizan un entorno en el cual se comunican en modo línea, es decir, la comunicación que se realiza es en ambas direcciones.

Posibilita la comunicación y que funcionen todos los controles, es decir, que asegure el envío de paquetes de datos, sincronizar las transmisiones y garantizar la recepción.

Otro punto importante, es la función de TCP encarga de “controlar” la velocidad de los datos, utilizando su capacidad para emitir mensajes con un tamaño variable, los cuales se conocen como segmentos.

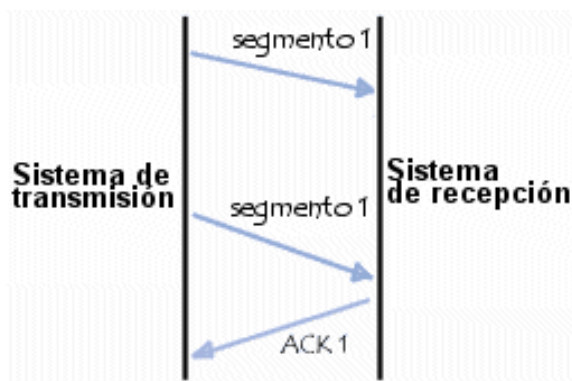


Figura 34, Transferencia en el protocolo TCP [9]

4.8.2 Servidor HTTP

El protocolo de transferencia HTTP es usado para la transferencia de información entre sistemas de forma clara y rápida. Entre sus ventajas podemos incluir que permite una serie de métodos para indicar la finalidad de la petición, cabeceras, obtener datos y/o pruebas de conexión.

Está basado en un paradigma de peticiones y respuestas, debido a que el cliente envía una petición en forma de método, URI y una versión de protocolo seguida de los modificadores de la petición en forma muy parecida a un mensaje MIME, la cual posteriormente será enviada y obtendrá un posible contenido de respuesta. El servidor, contesta a línea de estado que incluye la versión del protocolo y un código que indica si existió error o hubo éxito, seguido de la información del servidor en forma de mensaje MIME y su posible contenido-

Por lo general, el cliente es el que inicia la comunicación HTTP la cual consiste en la petición de algún recurso del servidor, lo cual se puede realizar de manera directa o por medio de intermediarios [10].

Para poder construir un servidor HTTP para que distintos dispositivos puedan comunicarse de manera directa a través del protocolo HTTP reside en que la aplicación móvil será la encargada de proporcionar los datos solicitados por las diferentes aplicaciones asociados, de modo que no sea necesario intermediarios.

Se proponen tres niveles, uno para poder realizar la configuración del servidor, en donde especificaremos el puerto por defecto, la cantidad máxima de procesos simultáneos que pueden ser atendidos, y el tiempo que puede pasar sin actividad una conexión abierta.

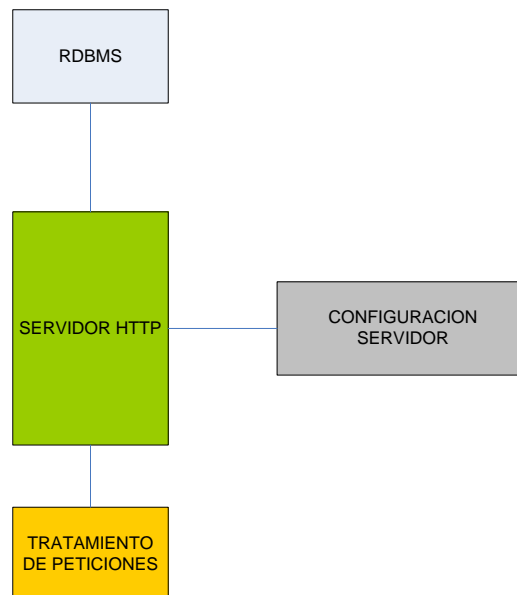


Figura 35, Arquitectura Servidor HTTP [11]

El siguiente nivel, será el encargado del tratamiento de peticiones el cual reconocerá las conexiones entrantes y llevara a cabo las tareas para poder resolver las peticiones HTTP que obtenga. Existen métodos soportados, los cuales son:

- GET: Solicitud de un documento o recurso que se encuentren en el sistema.
- HEAD: Información relativa de un recurso específico,
- OPTIONS: Métodos disponibles

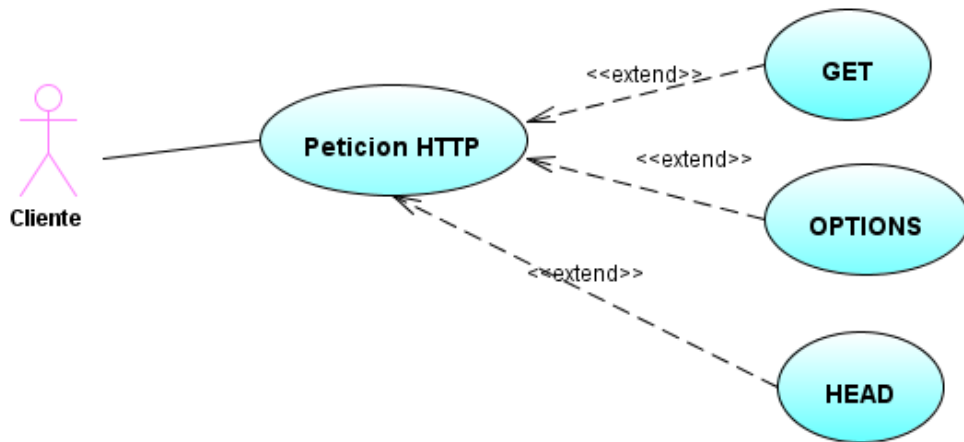


Figura 36, Caso de Uso de una Petición HTTP [11]

El último nivel, consiste en implementar herramientas que tengan la misión de poder obtener información que esta almacenada en la base de datos, de forma de un arreglo simple de bytes.

4.8.3 Servicios Web

Una de las principales ventajas de los Servicios Web es que describen una forma estándar para la integración de aplicaciones mediante el uso de **XML, SOAP, WSDL y UDDI** sobre alguno de los protocolos existentes en internet.

Para poder describir los datos se utiliza XML mientras que para poder transferir los datos se utiliza SOAP. Por otro lado, UDDI se encarga de describir los servicios actuales que están disponibles a diferencia de WSDL que solo es empleado para describir los servicios.

Desde sus inicios, ha permitido a las empresas intercambiar datos sin necesidad de conocer los detalles de los respectivos SIA de las empresas asociadas.

Es importante mencionar, que a diferencia de los modelos de tipo **Cliente/Servidor**, estos no otorgan una interfaz gráfica para el usuario. En vez de ellos, comparten todo lo asociado a la lógica del negocio, los procesos y los datos a través de una interfaz de programas en red. Esto quiere precisar, de que *conectan programas*, por lo tanto son programas que no *interactúan* con el usuario final de la aplicación. Es posible, si el desarrollador decide, poder agregar a los Servicios Web una interfaz para los usuarios.

Los Servicios Web permiten a distintas aplicaciones, de diferentes lugares, comunicarse entre ellos sin la necesidad de estar desarrollando programas extras, esto debido a que la comunicación se realiza utilizando XML.

Otra característica es que no están asociados a ningún SO o Lenguaje de Programación, esto quiere decir, como ejemplo, que si se tiene una aplicación desarrollada en Ruby esta podrá comunicarse con otra escrita en Java como también Aplicaciones Windows pueden establecer comunicación con entornos UNIX.

La popularidad ha ido creciendo y han estado continuamente mejorando los procesos de negocios.

A continuación, se describen los estándares de los Servicios Web con los cuales se han construido en conjunto los estándares para asegurar la operación continua como también la seguridad.

- XML: Es la especificación desarrollada por W3C la cual permite a los desarrolladores crear sus propios TAGS, los cuales ayudan a habilitar definiciones, transmisiones, validaciones y por sobre todo interpretar los datos entre las organización asociadas.
- SOAP: Es un protocolo para la mensajería construido en XML que es usada para poder codificar la información de los *requerimientos* y para responder a los mensajes antes de enviarlos por la RED. Estos mensajes utilizados en SOAP son independientes de los sistemas operativos y pueden ser transportados en diferentes protocolos de internet como lo es MIME, HTTP o SMTP.

- WSDL: También utiliza XML, se ocupa de poder definir a los Servicios Web como colecciones de punto de comunicación capaces de intercambiar mensajes. Es parte integral de UDDI y parte del registro global de XML.
- UDDI: Es el encargado de publicar los Servicios Web, de esta manera se vuelven públicos para que las demás empresas puedan utilizarlos.

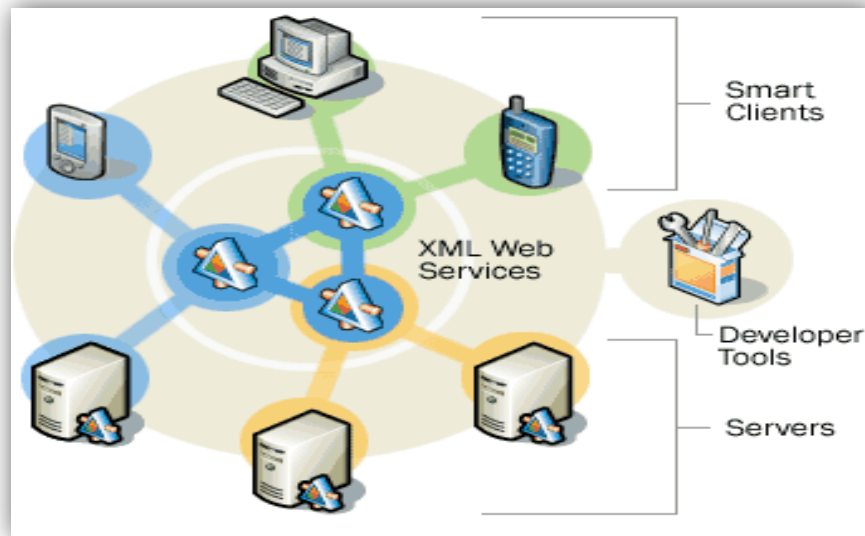


Figura 37, Arquitectura Servicio Web [13]

4.8.4 Tecnología a utilizar para la comunicación de datos

Debido a que un Servicio Web aporta *interoperabilidad* (habilidad de dos o más sistemas o componentes para intercambiar información y utilizarla la información intercambiada), junto con la cualidad de poder intercambiar información entre aplicaciones que se ejecutan en diferentes Sistemas operativos como también programadas en distintos lenguajes de programación y de fomentar los estándares y protocolos basados en texto que hacen más fácil acceder a los contenidos y funcionamiento es que se opta por utilizar esta tecnología.

Esto reduce los costos que supondría tener que realizar algún tipo de migración, como también aumentar la seguridad debido a que se aprovecha de los sistemas de seguridad (firewall) existentes en los SIA involucrados.

Para el desarrollo del proyecto se utilizará SOAP, el cual define dos objetos en diferentes procesos pueden comunicarse por media de intercambio de datos XML (mensajes).

Cada uno de los métodos invocará tareas específicas complejas para poder llevar a cabo las diferentes tareas y actividades necesarias para la aplicación móvil.

El siguiente diagrama presenta un ejemplo del uso de SOAP y el intercambio de mensajes.

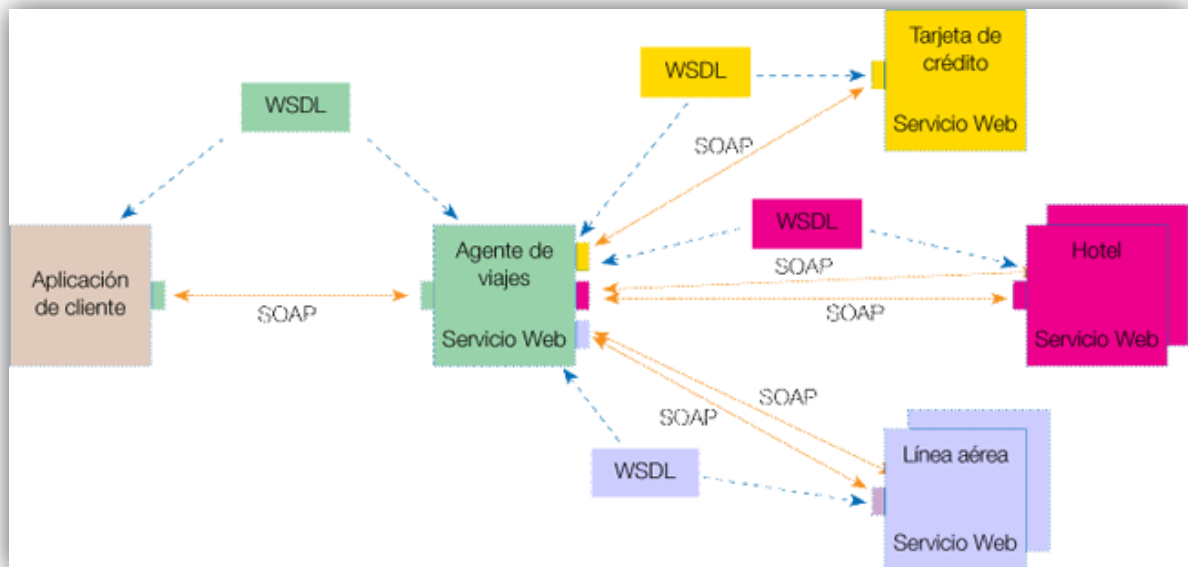


Figura 38, SOAP y los mensajes XML [14]

4.9 Factibilidad Legal

SertChile posee para su uso interno la licencia de **SQL Server 2005**, la cual permite tener instalado en un solo servidor el mencionado motor de base de datos.

El servidor está basado en IIS, del cual también se tienen las licencias de uso.

Por otro lado, Android y SOAP tiene las licencias de **Apache 2.0** y **GNU GPL 26**, las cuales permiten el desarrollo de aplicaciones de manera libre, sin necesidad de realizar algún tipo de pago por el entorno de desarrollo (SDK) o el uso de la aplicación en algún medio empresarial, siempre y cuando se mantenga la información referente a Android y Apache.

Como se maneja información de carácter importante, según la **Ley Chilena 19.223**, que declara lo siguiente:

Artículo 1°.- El que maliciosamente destruya o inutilice un sistema de tratamiento de información o sus partes o componentes, o impida, obstaculice o modifique su funcionamiento, sufrirá la pena de presidio menor en su grado medio a máximo.

Si como consecuencia de estas conductas se afectaren los datos contenidos en el sistema, se aplicará la pena señalada en el inciso anterior, en su grado máximo.

Artículo 2°.- El que con el ánimo de apoderarse, usar o conocer indebidamente de la información contenida en un sistema de tratamiento de la misma, lo intercepte, interfiera o acceda a él, será castigado con presidio menor en su grado mínimo a medio.

Artículo 3°.- El que maliciosamente altere, dañe o destruya los datos contenidos en un sistema de tratamiento de información, será castigado con presidio menor en su grado medio.

Artículo 4°.- El que maliciosamente revele o difunda los datos contenidos en un sistema de información, sufrirá la pena de presidio menor en su grado medio. Si quien incurre en estas conductas es el responsable del sistema de información, la pena se aumentará en un grado."

Y por cuanto he tenido a bien aprobarlo y sancionarlo; por tanto promúlguese y llévese a efecto como Ley de la república. 68

5 Capítulo quinto: Diseño

5.1 Arquitectura del Sistema

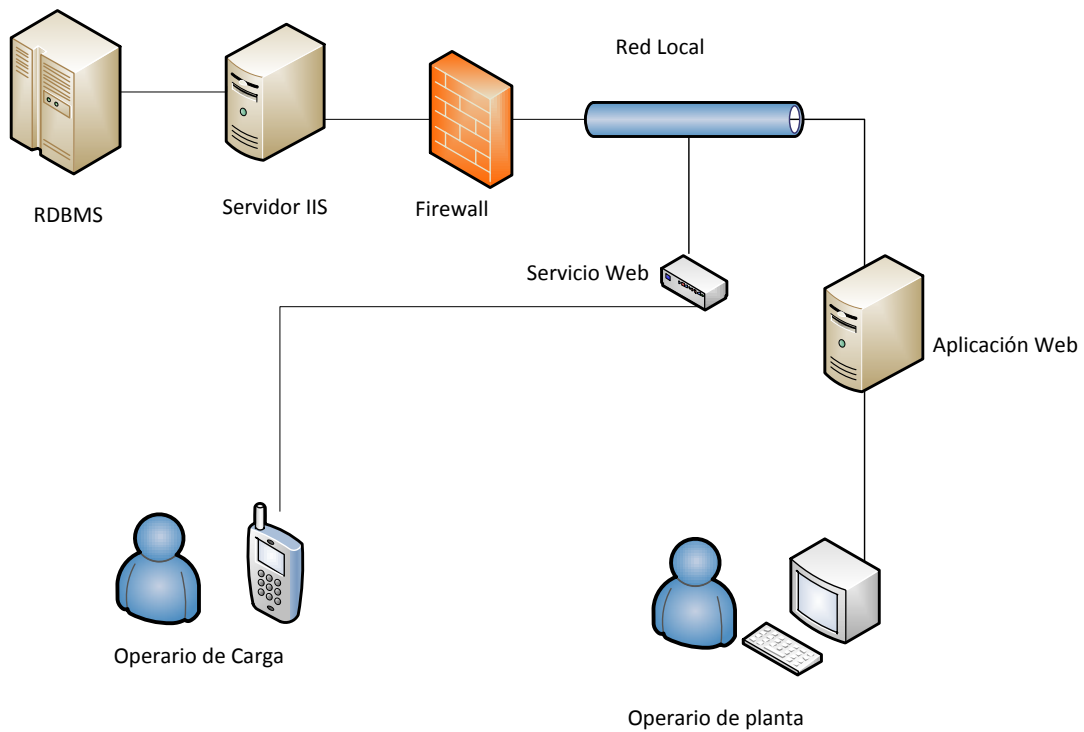


Figura 39, Arquitectura del Sistema

5.2 Modelo de Datos

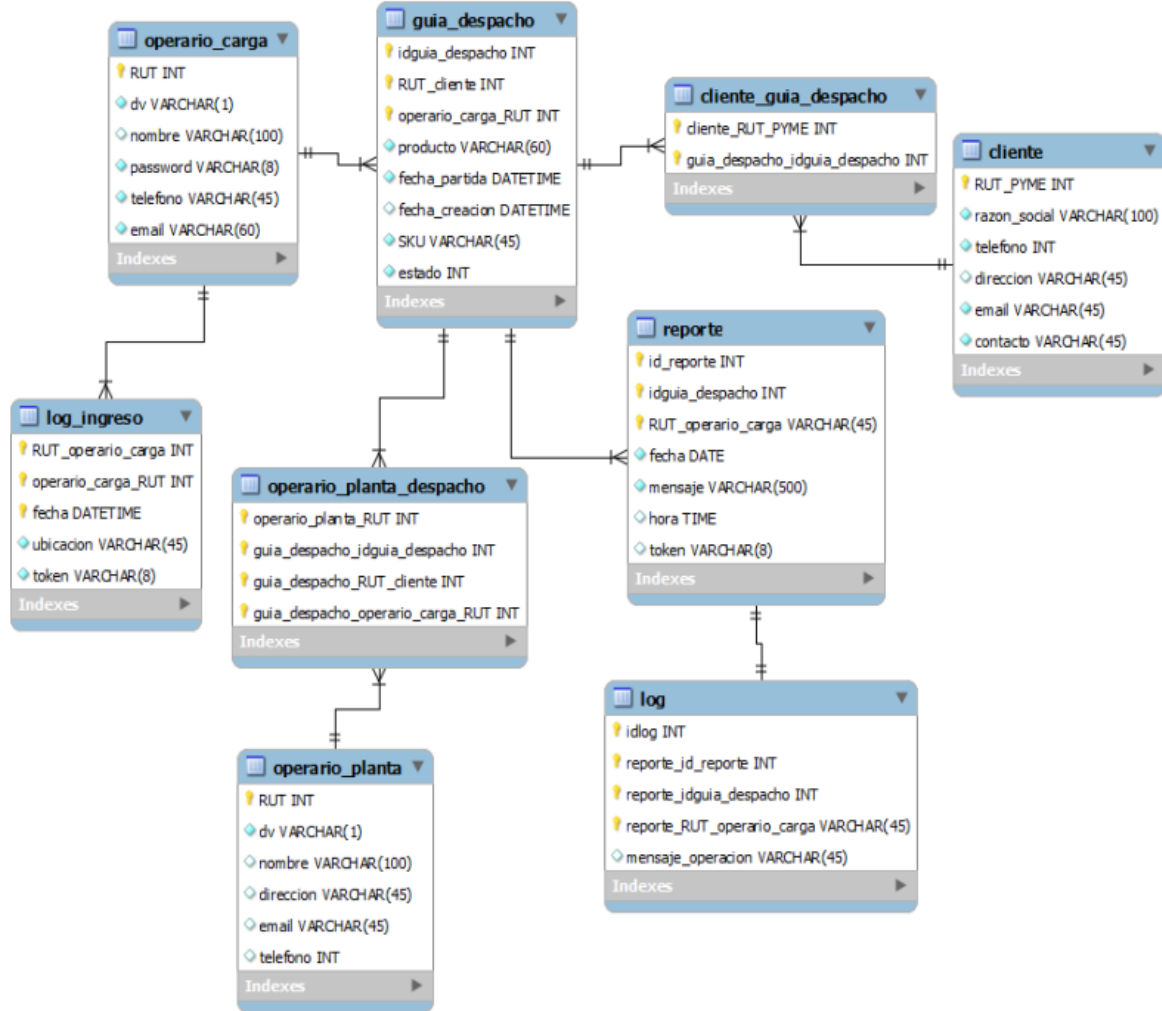


Figura 40, Modelo de datos propuesto

5.3 Modelo de datos actual (extracto)

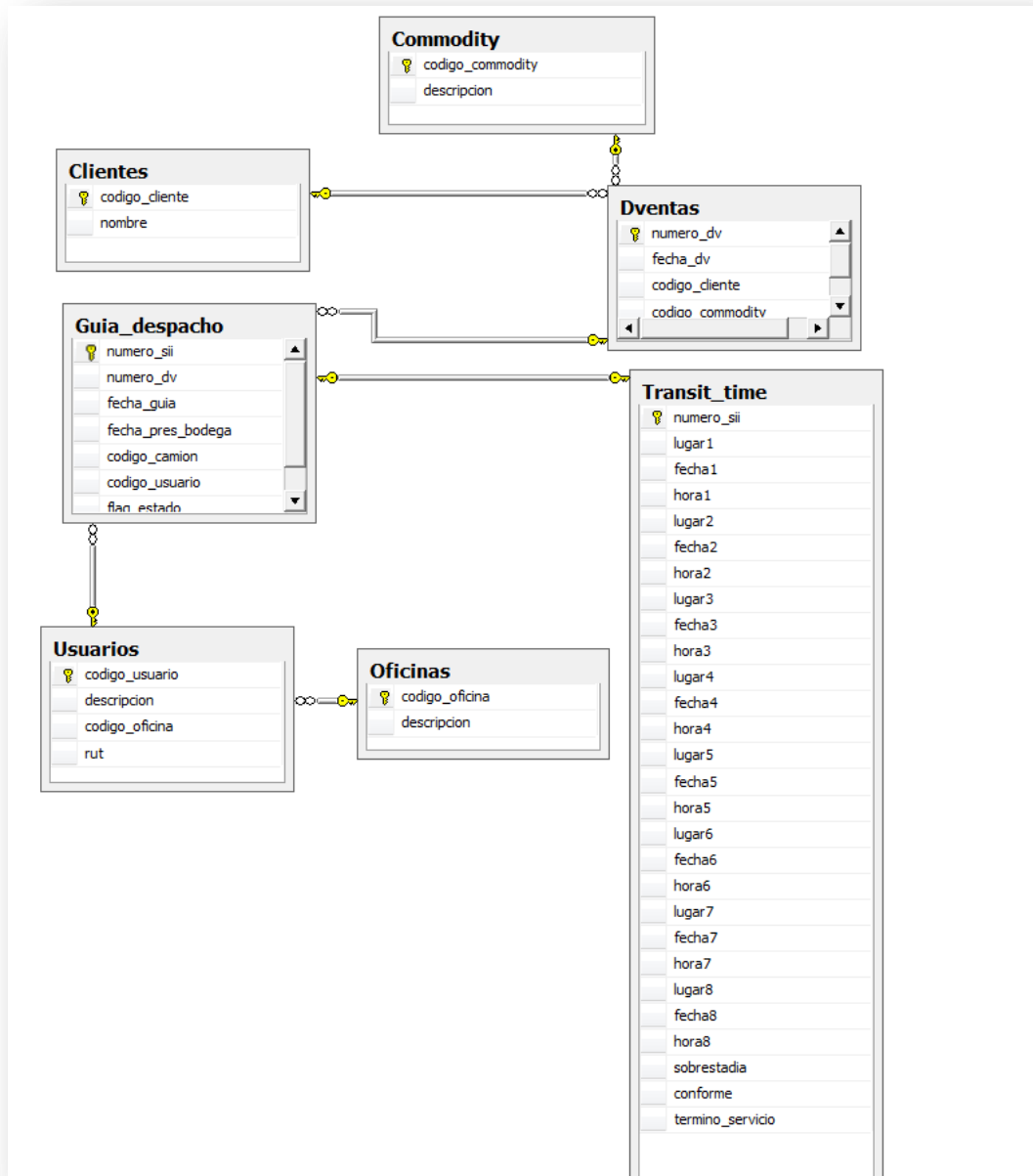


Figura 41, Modelo Relacional actual del ERP

5.3.1 Modificación al Modelo Relacional actual

Para poder considerar de manera adecuada los datos de geo localización (latitud - longitud) es necesario agregar una nueva tabla al modelo relacional actual del ERP de la empresa. La nueva tabla contendrá la información asociada al número de guía de despacho, el lugar del reporte (asociado al metadato de lugar-N), latitud y longitud.

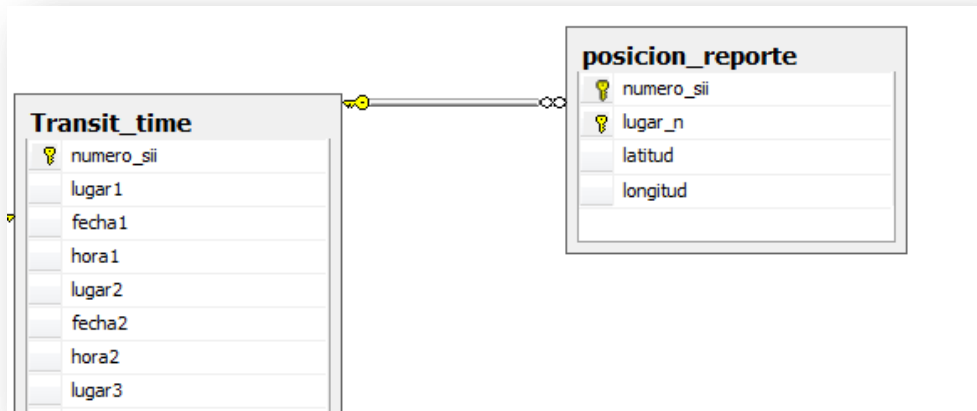


Figura 42, Relación de la nueva tabla Posición reporte

Para poder gestionar el ingreso desde la aplicación móvil por parte de los operarios de carga es necesario agregar una nueva tabla la cual tendrá el código de la patente, el nombre y el RUT del operario.

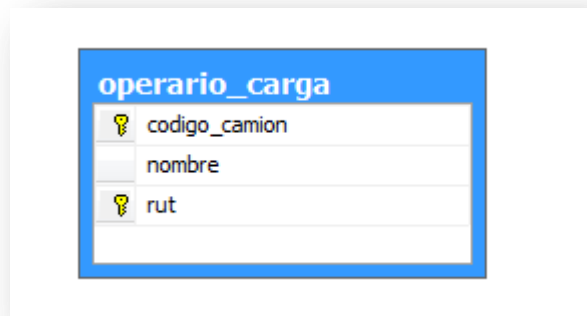


Figura 43, Nueva tabla Operario carga

5.4 Diseño Interfaces

5.4.1 Interfaz: Ingreso Operario de Carga



Figura 44, Interfaz Ingreso y validación

5.4.2 Interfaz: Operario de Carga - Historial Reporte y Pestaña General



Figura 45, Interfaz Historial Reporte y Pestaña General

5.4.3 Interfaz: Operario de Carga - Vista Principal – Pestaña Operario



Figura 46, Interfaz Vista General Pestaña Operario

5.4.4 Interfaz: Operario de Carga - Creación Reporte

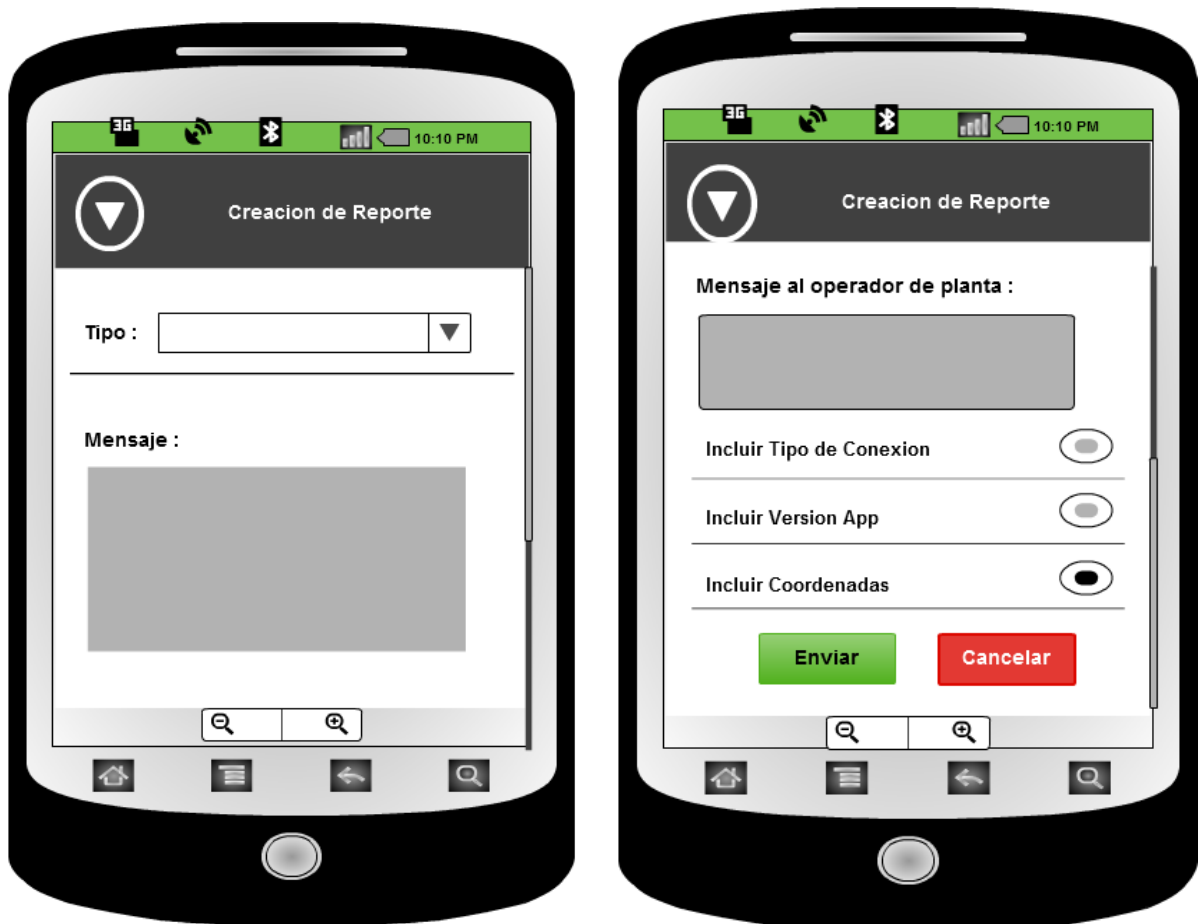


Figura 47, Interfaz Creación Reporte



Figura 48, Interfaz Creación Reporte Validación

5.5 Prototipo Funcional

5.5.1 Selección Aplicación desde el Menú



Figura 49, Selección aplicación

5.5.2 Ingreso de usuario (Operario de carga)



Figura 50, Prototipo Ingreso de usuario

5.5.3 Menú principal



Figura 51, Prototipo Menú principal

5.5.4 Menú principal – Envío de Reporte

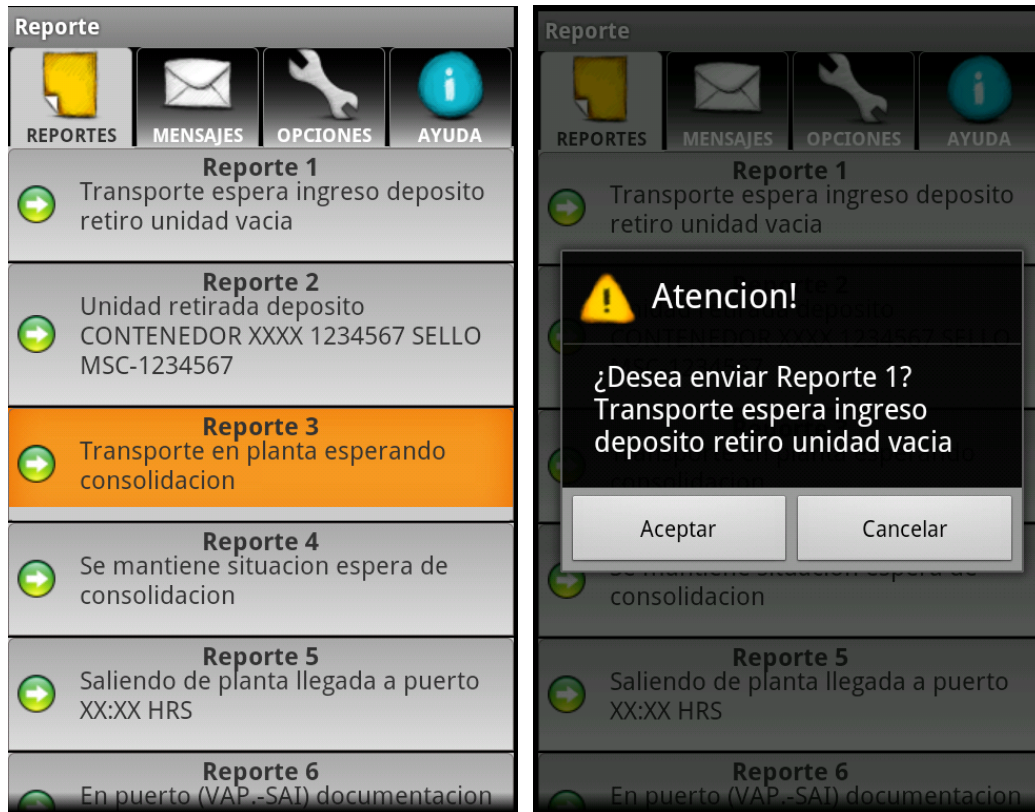


Figura 52, Prototipo Menú principal – Envío reporte

5.5.5 Menú principal – Pestaña Mensaje y opciones

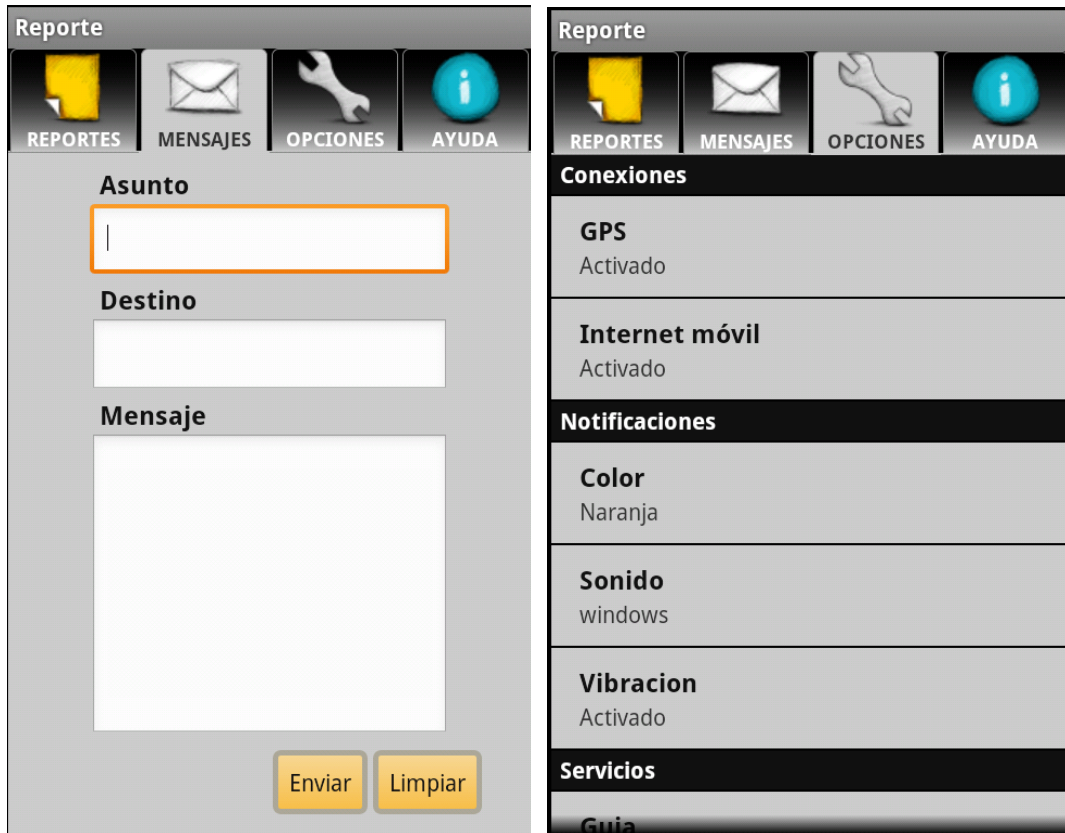


Figura 53, Prototipo Menú principal – Mensaje y opciones

5.5.6 Menú principal – Pestaña Ayuda



Figura 54, Prototipo Menú principal – Ayuda

5.6 Consideraciones para un óptimo desempeño

Uno de los problemas más comunes que existen con las aplicaciones en teléfonos móviles es la energía. Un uso no considerado de los recursos puede llevar a que la aplicación consuma toda la batería, como también una gran cantidad de recursos (memoria) del sistema operativo.

Entre los proveedores más habituales nos encontramos:

- **gps:** Este proveedor determina la localización mediante satélites. Dependiendo de las condiciones, este proveedor puede tomar un tiempo para volver una posición.
- **red:** Nombre del proveedor de ubicación de red. Este proveedor de ubicación determina según la disponibilidad de antena de telefonía móvil y puntos de acceso WiFi. Los resultados se recuperan por medio de una búsqueda de red.
- **pasiva:** Un proveedor de ubicación especial para la recepción de ubicaciones sin tener que iniciar una posición. Este proveedor se puede utilizar para recibir actualizaciones de localización pasiva cuando otras aplicaciones o servicios que lo soliciten sin tener que pedir a los lugares a ti mismo. Este proveedor devolverá ubicaciones generados por otros proveedores.

La siguiente tabla especifica lo que gastan algunos proveedores de Android en tareas fundamentales para este proyecto como lo son el GPS y tareas en segundo plano.

Efectividad	Uso batería	Tecnología
20ft	Alta	GPS autonomo
		1. Utiliza el GPS del dispositivo
		2. Linea de señal hacia los satelites
		3. Utiliza alrededor de 7
		4. Puede llegar a demorar mucho tiempo para obtener una actualizacion
		5. No funciona en construcciones altas
200ft	Media-Baja	GPS asistido (internet)
		1. Utiliza el GPS del dispositivo como tambien es asistido por la internet
		2.Utiliza un consumo muy bajo de energia
		3. Muy preciso
		4. Trabaja sin ninguna linea de señal en el dispositivo
		5.Depende del carrier y del soporte del telefono
5300ft/ 1mile	Baja	WiFi MACID
		1.Muy rapido, no requiere GPS en el dispositivo
		2. No utiliza mucha bateria del todo
		3. Posee muy poca efectividad, algunas veces puede tener efectividad alta

Tabla 29, Optimización recursos

5.7 Definición métodos Servicio Web

- **buscarGuiaActiva:** Obtiene el número de la Guia de despacho (si existe) para la fecha actual.
Utiliza como parámetro: código_camion.
- **insertarReporte:** Inserta un nuevo reporte en la tabla 'Transit_time'.
Utiliza como parámetro: numero_guia, data_reporte, código_camion.
- **loginOperario:** Verifica si el usuario que está accedendo existe en el sistema.
Utiliza como parámetro: rut, clave.
- **obtenerHoraActual:** Obtiene la fecha y la hora actual del servidor para poder realizar las operaciones básicas de la aplicación, tales como lanzar notificaciones en ciertos periodos de tiempo y buscar guias de despacho disponibles.
Utiliza como parámetro: Ninguno.
- **obtenerServicio:** Devuelve verdadero si el servicio está activo. Utilizado para emplear la mitigación en caso de perder la conexión con el servicio web.
Utiliza como parámetro: Ninguno.
- **totalReportes:** Devuelve la cantidad de reportes existentes en la metadata de la tabla 'Transit_time'.
Utiliza como parámetro: Ninguno.

6 Capítulo sexto: Pruebas

La *partición equivalente* es un método de prueba de **caja negra** que divide el campo de entrada de una aplicación en clases de datos de los que se pueden derivar casos de prueba. Un caso de prueba ideal descubre de forma inmediata una clase de errores, que de otro modo requerirían la ejecución de muchos casos antes de detectar el error genérico.

- Si una condición de entrada especifica un rango, se define una clase de equivalencia válida y dos inválidas.
- Si una condición de entrada requiere un valor específico, se define una clase de equivalencia válida y dos inválidas.
- Si una condición de entrada especifica un miembro de un conjunto, se define una clase de equivalencia válida y una inválida.
- Si una condición de entrada es lógica, se define una clase válida y una inválida.

Ingreso:

- 1) Entrada: Usuario “173545177” - Contraseña “5177”
Salida: Dar paso. Usuario y contraseña válidos.
- 2) Entrada: Usuario “88918622” - Contraseña “5k9l3”
Salida: No dar paso. No válido por contener caracteres “no numéricos” y más de 5 caracteres.
- 3) Entrada: Usuario “171121” - Contraseña “5177”
Salida: No dar paso. Usuario debe contener al menos 8 caracteres.
- 4) Entrada: Usuario “173545177” - Contraseña “”
Salida: No dar paso. Se debe especificar contraseña.
- 5) Entrada: Usuario “ ” - Contraseña “5177”
Salida: No dar paso. Se debe especificar usuario.
- 6) Entrada: Usuario “17KQ45177” - Contraseña “5177”
Salida: No dar paso. Usuario no valido por poseer caracteres “distintos a K”
- 7) Entrada: Usuario “ ” - Contraseña “ ”
Salida: No dar paso. Usuario y contraseña deben ser especificados.

Envío Reporte:

- 1) Entrada: Mensaje “Choque ruta 68.Retraso aprox. 2 horas”
Salida: Dar paso. Mensaje válido.
- 2) Entrada: Usuario “ ”
Salida: No dar paso. Se debe especificar un mensaje.

7 Capítulo séptimo: Conclusiones y Trabajo Futuro

De lo analizado y consultado se pudo desglosar una mayor cantidad de problemáticas y de requerimientos que no habían sido captados en las reuniones previas con el cliente.

Esta etapa final ya contempla una solución estable que se basa en ciertas especificaciones para asegurar la comunicación entre el cliente y el servidor de manera segura, rápida y por sobre todo diseñada para utilizar los recursos de la manera más eficaz.

Existieron grandes cambios desde las interfaces hacia los prototipos funcionales debido a la solicitud del cliente para omitir información que podía ser sensible, organización de elementos, colores, logo, etc.

También se revisó la solución actual del cliente (ERP) encontrando falencias, para lo cual se proponen los ajustes necesarios para no interferir en el normal funcionamiento de su software como también asegurar el acople de la aplicación móvil a la estructura de sus datos.

Se desarrolla una aplicación que es personalizable para el usuario final, especificando colores, sonidos y funcionalidades propias del equipo. Se agregan mensajes desplegables para ayudar al usuario a lo largo de su interacción con la aplicación, además de agregar un apartado específico de ayuda, el cual resume y explica la utilización de este.

Uno de los problemas iniciales del software era la comunicación. Desde ahora, se utiliza un servicio de contingencia en conjunto con un Servicio Web al momento de perder la comunicación y no poder enviar los datos hacia la base de datos. Este servicio se activa sin que el usuario se percate y va guardando la información hasta poder recuperar la conectividad.

La primera línea de continuación de este trabajo es el desarrollo de la aplicación en ambiente **iOS** y **BlackBerry**. Si bien en este trabajo se optó por desarrollar utilizando Android como plataforma, es relevante tener una versión disponible para estas versiones porque existe un grupo de operarios de carga que actualmente no tienen un dispositivo móvil (5%) lo cual los obliga a utilizar otros dispositivos móviles dispuestos por la Empresa. Más adelante podría existir la necesidad de expansión por parte de la empresa, lo cual generaría mayor cantidad de operarios de planta, los cuales podrían tener otros sistemas operativos móviles.

Otro punto importante a futuro es la “*Implementación de mensajería entre operario de planta y operario de carga*” para lograr una comunicación a tiempo real entre el transportista y el operario de carga. La aplicación actualmente sólo considera la información relacionada a los reportes que se manejan hacia los clientes y la información que ellos esperan recibir, pero poder lograr un flujo de comunicación extra entre los operarios ayudaría a guiar a los operarios de carga a encontrar mejores rutas, caminos alternativos en caso de emergencias, encontrar precios más económicos para los combustibles y locales para la reparación o compra de repuestos.

Implementar este mismo sistema en otra empresa de transporte puede ser complejo, debido a que la gran mayoría de las empresas prefiere optar por la implementación de sistemas GPS. Por otro lado, cada empresa tiene problemáticas diferentes por resolver y quizás la solución

propuesta en este trabajo de investigación sea muy breve o en otros casos muy extensa. En los casos en los cuales la problemática sea parecida es muy probable que los modelos de datos, las reglas de negocio y la maduración tecnológica sean muy diferentes lo cual obliga necesariamente a realizar cambios en las lógicas y los tiempos de trabajo de la aplicación. Los algoritmos de optimización de batería, de obtención de latitud y longitud, de mitigación son portables para cualquier aplicación móvil que necesite utilizar la API interna de Android para el manejo de GPS.

Otra línea de trabajo interesante para este desarrollo es la aplicación de la “*Usabilidad en las aplicaciones móviles*” para poder dar una visión más práctica del sistema. Por lo general este tipo de aplicaciones tiene como usuarios finales a personas que no están muy familiarizadas con temas tecnológicos, por lo cual, realizar un análisis detallado de la usabilidad sería muy útil para el uso y el buen manejo de esta.

Para toda empresa de transporte la expansión es un hecho muy probable. Debido a esto, optimizar el sistema para una concurrencia mayor a esta podría ser un trabajo a desarrollar. Actualmente, las operaciones en la base de datos ya han sido optimizadas, bastaría con evaluar y diseñar los posibles problemas de comunicación derivados de la excesiva carga de usuarios.

8 Capítulo octavo: Referencias y Bibliografía

- [1] Reuters. *Google's Android leads U.S. Smartphone*. 4 Agosto 2010. Consultado 03/07/2012
<http://ca.reuters.com/article/businessNews/idCATRE6734HB20100804>
- [2] Laurence Goasduff. *Gartner Says Worldwide Smartphone sales soared in fourth Quarte of 2011 with 47 percent growt*. 2012. Consultado 13/07/2012
<http://www.gartner.com/it/page.jsp?id=1924314>
- [3] José García Santesmases. *Arquitectura y especificaciones Java Micro Edition*. 2006. Consultado 22/07/2012
http://grasia.fdi.ucm.es/j2me/_J2METech/index.html
- [4] Android Developers. *Conociendo Android, Arquitectura y detalle*. 2010. Consultado 13/04/2012
<http://www.android.com/about/ice-cream-sandwich/>
- [5] Android Developers, *Platform Versions and Current Distribution*. 2010. Consultado 09/04/2012
<http://developer.android.com/resources/dashboard/platform-versions.html>
- [6] BGR, *Details on the Windows Phone 7 architecture leaked*. 2010. Consultado 24/05/2012
<http://www.bgr.com/2010/04/19/details-on-the-windows-phone-7-architecture-leaked/>
- [7] Arstechnica, *Leaked Windows Phone 7 docs describe update, customization*. 2010. Consultado 13/04/2012
<http://arstechnica.com/microsoft/news/2010/04/leaked-windows-phone-7-docs-show-off-wince-6-underpinnings.ars>
- [8] Microsoft Corporation. *SQL Server 2005, Microsoft User Guide*, páginas 103-305. 2006. Consultado 03/05/2012
<http://www.microsoft.com/sqlserver/en/us/default.aspx>
- [9] Pedro J. Ponce de León. *Protocolo de control de transmisión, "DARPA INTERNET PROGRAM" – Especificación de Protocolos*, página 37-52. Marzo 2002. Consultado 23/09/2012
<http://www.rfc-es.org/rfc/rfc0793-es.txt>
- [10] José Alvares, *Protocolo HTTP*. 2001. Consultado 04/05/2012
http://www.uhu.es/josel_alvarez/NvasTecnProg/recursos/ProtocoloHTTP.pdf
- [11] Guillermo Diez-Andino Sancho. *Desarrollo de un servidor HTTP para dispositivos móviles, Universidad Carlos de Madrid.*, páginas 3-8. 03 Marzo 2003. Consultado 19/08/2012
- [12] Mario Saffirio. *¿Qué son los Servicios Web?*. 2006. Consultado 26/06/2012

<http://msaffirio.wordpress.com/2006/02/05/¿que-son-los-web-services/>

[13] Silvana Pacheco, *Servicios Web para recuperación de Información del Sistemas de Gestión Académica*, Universidad Nacional De Loja. Páginas 101-155. 2002. Consultado 26/06/2012

[14] Austin Rasmussen, *Android Lists IV: Accessing and Consuming a SOAP Servicio Web*. 24 Mayo 2012. Consultado 26/06/2012

<http://android.vexedlogic.com/2011/04/17/android-lists-iv-accessing-and-consuming-a-soap-web-service-i/>