UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA

PERFIL DE PROYECTO DE GRADO

TITULO: **"APLICACIÓN ANDROID PARA LA GESTION Y CONTROL DE LA ENTREGA Y RECOJO DE CORRESPONDENCIA CASO: TNT EXPRESS"**

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMATICA

MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: Marco Antonio Hinojosa Fernandez

TUTOR METODOLOGICO: Lic. Freddy Miguel Toledo Paz

ASESOR: Lic. Marcelo Aruquipa Chambi

LA PAZ – BOLIVIA

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA

LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

DEDICATORIA

Dedico este trabajo primeramente a mis Padres Julian Hinojosa y Gregoria Fernandez por brindarme siempre su apoyo y cariño a lo largo de toda mi vida para apoyarme en todas las formas que pudieron; también a mis hermanos Cinthya y Eduardo que los llevo siempre en mi corazón junto a todos mis sobrinos.

También dedico este logro a mi querida esposa Leoni Espinoza y mi Hija Aylin Giselle por ser el motor y mi razón de ser, por apoyarme y motivarme a terminar este trabajo.

AGRADECIMIENTOS

Agradecer primero a Dios por todas las bendiciones recibidas en mi vida.

Agradecer a mi Tutor Lic. Freddy Miguel Toledo por el tiempo dedicado en su materia y brindarme la oportunidad de poder presentar este trabajo, le deseo todo el éxito del mundo en sus actividades que realiza.

Agradecer a mi Revisor Lic. Marcelo Aruquipa por las correcciones y observaciones a lo largo del proceso de la elaboración de este trabajo, le deseo éxitos a lo largo de su vida.

Agradecer a mi querida Carrera de Informática y a sus autoridades porque gracias a ella vivo muchas experiencias profesionales y en verdad es lo que me gusta hacer.

Agradecer a la empresa TNT Intertrade Express SRL por la oportunidad de desarrollar este proyecto y llevarlo a cabo hasta su implementación.

Correo: marco.hinojosa@outlook.com

RESUMEN

La nueva tendencia de la tecnología es dar a los usuarios la posibilidad de tener todo lo que pueden llegar a necesitar en un dispositivo que brinde “movilidad”, definida como la capacidad de acceder a la información y a los servicios en cualquier momento, de cualquier manera y en cualquier parte.

El problema principal en la empresa es no contar con una aplicación móvil que permita la gestión y control del servicio de recojos, entrega de envios y correspondencia apoyada en un modelo de ordenamiento y priorización, además que permita actualizar y brindar información oportuna con pruebas de entregas digitales y en línea a los clientes.

El presente proyecto tiene como objetivo desarrollar una herramienta en Android que permita ayudar a actualizar la información de las entregas de correspondencia y optimizar los tiempos de respuesta en los estatus de cada envío y tener la información de forma oportuna.

El proyecto será de beneficio para la parte operativa y gerencial, colaborando en el desempeño de las tareas gestión en la entrega y recolección con la actualización y registro de los envios que se tenga pendientes, además de controlar el rendimiento de cada mensajero.

INDICE GENERAL

CAPITULO 1 1
1.1 Introducción 1
1.2 Antecedentes 1
1.3 Planteamiento Del Problema 2
1.3.1 Problema General 2
1.3.2 Problema Específicos 2
1.4 Hipótesis 3
1.5 Objetivos 3
1.5.1 Objetivo General 3
1.5.2 Objetivos Específicos 3
1.6 Justificación 4
1.6.1 Justificación Social 4
1.6.2 Justificación Técnica 4
1.6.3 Justificación Económica 4
1.7 Límites y Alcances 4
1.7.1 Limites 4
1.7.2 Alcances 4
CAPITULO 2 6
2 Marco Teórico 6
2.1 Introducción 6
2.2 Marco institucional - TNT Express 6
2.2.1 Misión 7
2.2.2 Visión 7
2.2.3 Servicios 7
2.2.4 Courier 8
2.2.5 Courier Nacional 8
2.2.6 Proceso del servicio de Courier Nacional 8
2.2.7 Definición de Logística 9
2.2.8 Guías 10
2.2.9 Recolección 11

2.2.10 Entregas 12
2.2.11 Prueba de entrega 12
2.2.12 Manifiestos 12
2.2.13 Rastreo y Seguimiento 13
2.3 Tecnologías de desarrollo 13
2.3.1 Sistemas operativos móviles 13
2.3.1.1 Sistema operativo móvil Android 15
2.3.1.2 Arquitectura de Android 16
2.3.2 Web API 17
2.3.3 Código QR 18
2.3.4 Arquitectura Cliente – Servidor 18
2.3.5 Programación En Tres Capas 18
2.3.6 PHP 19
2.3.7 MySQL 19
2.3.8 Ajax 19
2.3.9 Android Studio 19
2.3.10 JDK Java 19
2.3.11 Microsoft Visio 20
2.3.12 Adobe Dreamweaver 20
2.4 Teoría de colas 20
2.4.1 Colas 21
2.4.2 TDA (Tipo de Dato Abstracto) 21
2.4.3 TDA Cola de Prioridad 21
2.4.4 Heaps 21
2.5 Metodologías agiles 22
2.5.1 Metodologías agiles para el desarrollo de software móvil 23
2.5.2 Características y Requerimientos Específicos Del Entorno Móvil 23
2.6 Mobile – D 24
2.7 Programación extrema XP 26
2.8 Valores en la programación extrema (XP) 28
2.8.1 Simplicidad 28
2.8.2 Comunicación 28

2.8.3 Retroalimentación (FeedBack) 29
2.8.4 Coraje o valentía 29
2.8.5 Respeto 29
2.9 Fases programación extrema 29
2.9.1 Planificación 30
2.10 Diseño del sistema 33
2.11 Codificación y Desarrollo 34
2.12 Pruebas 35
2.13 Modelos de negocio 36
2.14 Leguaje Unificado de Modelado - UML 36
2.14.1 Casos de Uso 37
2.15 Métricas de Calidad de Software 37
2.15.1 Modelo de Mc Call 37
2.16 COCOMO (COnstructive COst MOdel) 38
2.16.1 Justificación de los valores 40
2.16.2 Atributos de software 40
2.16.3 Atributos de hardware 41
2.16.4 Atributos del personal 41
2.16.5 Atributos del proyecto 41
2.16.6 Cálculo del esfuerzo del desarrollo: 41
2.16.7 Cálculo tiempo de desarrollo: 42
2.16.8 Productividad: 42
2.16.9 Personal promedio: 42
2.17 Seguridad 42
2.17.1 Encriptación MD5 42
2.17.2 Servicios de seguridad básicos 42
2.17.3 Autenticación de los interlocutores 43
2.17.4 Autorización 43
CAPITULO 3 44
3 Diseño del Sistema 44
3.1 Fase de exploración 44
3.1.1 Descripción de usuarios 44

3.1.2 Rational Unified Process RUP – UML 45
3.1.2.1 Diagrama de Contexto 45
3.1.1.1 Diagrama de Casos de Uso 45
3.1.3 Requerimientos iniciales.....46
3.1.4 Análisis de requerimientos iniciales.....46
3.1.5 Limitaciones.....48
3.1.6 Supuestos y dependencias 48
3.2 Fase de Inicialización 49
3.2.1 Establecimiento del proyecto 49
3.2.2 Configuración del ambiente de desarrollo.....49
3.2.2.1 Configuración de servidor web 49
3.2.2.2 Configuración de App móvil.....49
3.2.3 Entrenamiento.....49
3.2.4 Planificación inicial 49
3.2.4.1 Análisis de requerimientos y pre-requisitos 50
3.2.4.2 Gestión de recojos.....50
3.2.4.3 Gestión de entregas.....50
3.2.4.4 rastreo de guias 50
3.2.4.5 Noticias corporativas 50
3.2.5 Planificación de fases.....50
3.3 Fase de Producción.....52
3.3.1 Diseño del Sistema 52
3.3.2 Historia de usuarios 53
3.3.3 Detalle de las Historias de usuario.....58
3.4 Fase de Estabilización y Fase de Pruebas – Corrección.....58
3.4.1 Tarjeta CRC 59
3.4.2 Casos de Prueba 59
3.4.3 Interfaz de usuario.....59
3.4.3.1 Login del sistema 59
3.4.3.2 Menú principal 61
3.4.3.3 Modulo noticias.....64
3.4.3.4 Modulo mis mapas 66

3.4.3.5 Modulo rastreo de guia 68
3.4.3.6 Modulo mis recojos 70
3.4.3.7 Modulo mis entregas 73
CAPITULO 4 77
4. Seguridad, Control y Análisis Costo/Beneficio 77
4.1 Seguridad 77
4.2 Identificación y Autenticación 77
4.3 Encriptación con MD5 78
4.4 Auditoria 78
4.5 Análisis Costo-Beneficio del Sistema 79
4.6 Modelo Base de Estimación COCOMO 79
4.6.1 Punto Objeto 79
4.7 Beneficios 83
CAPITULO 5 85
5. Conclusiones y Recomendaciones 85
5.1 Conclusiones 85
5.2 Recomendaciones 85
Bibliografía 87

INDICE DE TABLAS

Tabla 2.1: Estructura de una Guia 11
Tabla 2.2: Comparativa de las principales plataformas móviles 14
Tabla 2.3: Formas de implementar una cola de prioridad 21
Tabla 2.4: Coeficiente para tipos de proyecto 39
Tabla 2.5: Conductores de Costo 39
Tabla 3.1: Descripción de usuarios 44
Tabla 3.2: Descripción de usuarios 47
Tabla 3.3: Planificación de fases 50
Tabla 3.4: Plantilla de historia de usuario 53
Tabla 3.5: Historia de usuarios: Listado de entregas 53
Tabla 3.6: Historia de usuario: Confirmar entrega 54
Tabla 3.7: Historia de usuario: Listado de recojos 55
Tabla 3.8: Historia de usuario: Confirmar recojos 56
Tabla 3.9: Historia de usuario: Rastreo de guia 57
Tabla 3.10: Historia de usuario: Login e ingreso al sistema 57
Tabla 3.11: Detalles de historia de usuarios y tareas 58
Tabla 3.12: Login del sistema 59
Tabla 3.13: Menu principal 61
Tabla 3.14: Modulo de noticias 64
Tabla 3.15: Modulo mis mapas 66
Tabla 3.16: Modulo rastreo 68
Tabla 3.17: Modulo mis recojos 70
Tabla 3.18: Modulo mis entregas 73
Tabla 4.1: Tabla de ponderación 79

INDICE DE FIGURAS

Figura 2.1: Logistica en TNT Express.....10
Figura 2.2: Porcentaje de teléfonos inteligentes vendidos según su sistema operativos hasta el segundo semestre de 2015 en el mundo 13
Figura 2.3: Arquitectura de Android.....16
Figura 2.4: Fases de Mobile-D 25
Figura 2.5: Proceso XP.....27
Figura 2.6: Fases de la Programación Extrema 29
Figura 2.7: Historia de usuarios.....31
Figura 2.8: Proceso de análisis en programación extrema 31
Figura 2.9: Proceso de Desarrollo con “User-Stories” 32
Figura 2.10: Seguridad punto a punto frente a la seguridad extremo a extremo f 43
Figura 3.1: Fases de Mobile-D 44
Figura 3.2: Diseño general del sistema.....52
Figura 4.1: Seguridad del Sistema 78

CAPITULO 1

1.1 Introducción

El auge de los servicios que se proveen hoy en día en Internet, el avance tecnológico, la tendencia hacia dispositivos más pequeños y más rápidos, junto con la necesidad de acceso a la información en cualquier momento, son los factores determinantes del surgimiento de nuevas tecnologías en el desarrollo de todo tipo de sistemas y aplicaciones para teléfonos celulares, los Smartphones, las tablets PC y ahora los relojes inteligentes PDA, del inglés Personal Digital Assistant (Asistente Digital Personal).

El desarrollo de estas aplicaciones conlleva una constante renovación y actualización de conocimientos específicos en nuevas tecnologías, requiriendo profesionales siempre actualizados y con acceso constante a nuevos dispositivos, para que puedan ofrecer soluciones innovadoras y eficaces a los problemas, requerimientos y necesidades que las empresas deben afrontar.

La nueva tendencia de la tecnología es dar a los usuarios la posibilidad de tener todo lo que pueden llegar a necesitar en un dispositivo que brinde "movilidad", definida como la capacidad de acceder a la información y a los servicios en cualquier momento, de cualquier manera y en cualquier parte.

Actualmente muchas empresas de servicios de Courier y mensajería express requieren mantener informados a sus clientes del estado, la ubicación y prueba de entrega de sus envios. Ahora bien, algunas de estas empresas optan por la actualización de sus envios manualmente en hojas y las más avanzadas realizan esta tarea con dispositivos específicos como scanner en computadoras portátiles de última tecnología los cuales almacenan los datos de entrega para ser cargados posteriormente en sus sistemas. El costo elevado de estos dispositivos recae en las empresas asociadas a estas grandes multinacionales donde por el tipo de movimiento local no justifica estos gastos.

El presente proyecto tiene como objetivo desarrollar una herramienta que permita ayudar a actualizar la información de las entregas de correspondencia y optimizar los tiempos de respuesta en los estatus de cada envío y tener la información de forma oportuna.

1.2 Antecedentes

Los últimos avances y desarrollos tecnológicos ocurridos en estos últimos años han logrado diseñar y crear aplicaciones móviles nunca antes pensadas sino con ayuda de tecnología específica y

costosa. Estas circunstancias traen consigo nuevas oportunidades de negocio y nuevas aplicaciones innovadoras utilizadas como herramientas de trabajo empresarial, educativo, emprendedor y científico.

En empresas multinacionales del servicio de mensajería courier como es el caso de DHL, FEFEX, UPS y TNT ya cuentan desde hace años con scanner móviles donde les permite la actualización de recojos y entregas de envios, la empresa Honeywell es una de la más importantes en proveer este tipo de herramientas que tienen un alto costo en cuanto a licencias y dispositivos específicos.

Existen aplicaciones móviles en diferentes plataformas como Android, IOS y Windows donde brindan el servicio de consultas, rastreo, cotizaciones y actualización de servicio de correspondencia, las cuales están diseñadas y adaptadas al tipo de mercado específico para cada empresa en particular de cada país.

Por el momento no existen proyectos desarrollados que están dirigidos al servicio de mensajería Express, y correspondencia, los cuales se adapten al tipo de mercado nacional y que estén operando para este fin.

1.3 Planteamiento Del Problema

El proceso de entrega de la correspondencia parte de la distribución y separación por zonas dentro de la ciudad, los cuales son asignadas de forma manual a los mensajeros que cumplirán con esta tarea, donde cada uno debe consolidar y registrar todos sus envios en una hoja de manifiesto y priorizar los que vengan con el servicio de Prioridad y los que son urgentes de entrega como los servicios Expresos. De esta manera cada mensajero tiene la tarea de informar al final del día o mediante llamada telefónica si la entrega fue hecha. También así llevar un control de que guias quedan pendientes de entrega para ser coordinadas con mejores datos y actualizar la información en la página web de TNT para el rastreo de los clientes.

1.3.1 Problema General

El problema principal en la empresa es no contar con una aplicación móvil que permita la gestión y control del servicio de recojos, entrega de envios y correspondencia apoyada en un modelo de ordenamiento y priorización, además que permita actualizar y brindar información oportuna con pruebas de entregas digitales y en línea a los clientes.

1.3.2 Problema Específicos

- Falta de un modelo para gestionar la entrega de envios en base a la prioridad de cada uno de ellos tomando en cuenta el tiempo de entrega, tipo de servicio y tipo de cliente.

- Falta de herramientas para que cada mensajero pueda consultar sus recojos y entregas que debe realizar y tiene asignado en el día, además de las que quedaron pendientes anteriormente.
- Pérdida de tiempo en la comunicación con la oficina central para consultar los recojos programados en el día además de solicitar información de cada uno de ellos.
- Demora en la actualización del estado cada envío, y actualización de datos de entrega.
- Demora por parte del supervisor en la coordinación con cada mensajero para la asignación de entregas de envios.
- Falta de reportes que ayuden a observar el rendimiento de cada mensajero según el número de entregas efectuadas al día.
- Ilegibilidad de los datos del consignatario en las copias físicas de las pruebas de entrega.
- Verificación y control de que el mensajero cumplió con su ruta de recolección y entregas.
- Costos elevados para adquirir dispositivos scanner que permiten el registro de entregas y recojos de los envios, junto al software no adaptado y diseñado para el tipo de mercado nacional.

1.4 Hipótesis

La aplicación móvil en Android ayudara a realizar el seguimiento de las entregas y recojos de la correspondencia mediante un modelo de priorización manteniendo informado y actualizado el estado de cada envío y obtener pruebas de entrega en línea.

1.5 Objetivos

1.5.1 Objetivo General

Gestionar y controlar la entrega de correspondencia mediante una aplicación móvil apoyada en un modelo de ordenamiento y priorización de toda la correspondencia que permita la actualización de cada envío para brindar información oportuna a los clientes y obtener datos digitales de pruebas de entrega.

1.5.2 Objetivos Específicos

- Controlar las entregas de correspondencia aplicando el modelo de colas de prioridad.
- Implementar y adecuar el modelo de colas con prioridad para realizar consultas para las entregas de todos los envios según al tiempo, tipo de servicio y/o tipo de cliente.
- Implementar en la aplicación móvil la lectura de código de barras de la guia y confirmación de entrega tomando una fotografía de la guía o la firma electrónica del consignatario que serán actualizados para ser consultado por el número de guía.

- Reducir el tiempo de respuesta en consultas con disponibilidad de información actualizada de cada envío desde internet.
- Agilizar el proceso de actualización de cada envío con el estatus que corresponde a cada situación que se presenta en cada entrega.
- Controlar el acceso al sistema Android mediante id personales para cada Courier.
- Generar consultas según criterio de propiedades de cada tipo de envío.

1.6 Justificación

1.6.1 Justificación Social

El proyecto será de beneficio para la parte operativa y gerencial, colaborando en el desempeño de las tareas gestión en la entrega y recolección con la actualización y registro de los envios que se tenga pendientes, además de controlar el rendimiento de cada mensajero.

1.6.2 Justificación Técnica

En la parte técnica se cuenta con el software para el desarrollo del sistema y en cuanto a los dispositivos móviles con requisitos básicos de conexión a datos móviles, cámara y la versión de Android desde la 4.1 Jelly Bean por ser una versión estándar. Además, se cuenta con la predisposición de la empresa para implementar el sistema utilizando herramientas y metodologías para el análisis y diseño de sistemas. También se dispone de un Servidor Web para la implementación de la Base de datos y conexión.

1.6.3 Justificación Económica

En cuanto a la situación económica del proyecto plantea un sistema que reduzca los gastos en adquirir dispositivos costosos y software con licencia de las empresas que ofrecen este servicio, además el sistema funcionara en cualquier dispositivo móvil con requisitos básicos de funcionamiento.

1.7 Límites y Alcances

1.7.1 Limites

El sistema será implementado a nivel nacional por tratarse de una herramienta fácil de utilizar y que servirá de control de todos los envios que se realiza todas las ciudades principales de cada departamento del país al estar conectado a una base de datos centralizada.

1.7.2 Alcances

Los alcances que tendrá la aplicación móvil principalmente serán:

- Gestión de entregas de los envios según clasificación por ciudad, mensajero, zona y tipo de servicio.
- Control de entregas de correspondencia donde se debe cumplir cierto tiempo en cuanto a la prioridad de cada uno de ellos.
- Lectura de código de barras y QR para realizar la actualización de cada envío.
- Prueba de entrega con la firma electrónica o la fotografía de la guía firmada o sellada.

CAPITULO 2

2 Marco Teórico

2.1 Introducción

Una empresa de mensajería o Courier es una empresa privada de transporte que dispone de vehículos de transporte, de transportistas y de una red propia o con franquicia de centros repartida por todo un territorio, desde los cuales se gestiona el envío y entrega de paquetes, cartas y documentos, que le son confiados por sus clientes (particulares y empresas) con distintas tarifas, en función de la urgencia con que haga sus entregas y del tamaño volumétrico de los envíos (volumen + peso).

Es muy común en el campo de las ventas a distancia, y en particular en las compras online, que las tiendas online recurran a los servicios de las empresas de mensajería para hacer llegar al domicilio de los compradores sus pedidos online. El resguardo de entrega del producto es el documento de la empresa de mensajería que firmamos cuando ésta nos hace entrega de un "paquete" en nuestro domicilio, del cual nos deja una copia. Este resguardo acredita que se nos ha hecho entrega de un paquete proveniente del vendedor o remitente.

2.2 Marco institucional - TNT Express

La compañía de TNT comenzó a ofrecer servicios de carretera y ferroviarios de mercancías en toda Australia incluyendo, por primera vez, nuevos servicios durante la noche. En 1958, la empresa llegó a ser conocido como **Thomas Nationwide Transport** o **TNT**, en 1961 se había convertido en TNT tanto éxito que fue listada en la bolsa australiana **(TNT Web, 2016)**.

TNT Express. Es una empresa global de entrega de correo y servicios de empresa con sede en Hoofddorp, en los países bajos. TNT opera como servicio postal nacional bajo el nombre de TNT Post, y en Latinoamérica con asociados en cada país, en Bolivia se maneja con la franquicia representado con la empresa nacional TNT Intertrade Express SRL.

La Vision de empresa es: *"Nuestra estrategia se fundamenta en ofrecer al mercado soluciones innovadoras, personalizadas y apoyadas por herramientas tecnológicas y de comercio electrónico de última generación. A esta estrategia une su filosofía de 'enfoque total al cliente', orientada a*

exceder en todo momento las expectativas de sus clientes, sus empleados, sus proveedores y la sociedad en general. Ninguna compañía realiza entregas en tantos destinos, en tan poco tiempo y con las máximas garantías de rapidez y fiabilidad. Parte fundamental de nuestra estrategia está concentrada en nuestras redes de distribución, donde disponemos de la red de transporte aéreo y terrestre más avanzada de Europa" **(TNT News, 2015)**.

2.2.1 Misión

Conectar negocios, mercados y gente de manera sustentable. Superar las expectativas de los clientes al transferir sus productos y documentos alrededor del mundo. Valoramos a nuestros clientes, brindándoles las soluciones más confiables y eficientes a través de redes de entrega. Tratamos de liderar la industria inculcando el orgullo en nuestra gente, creando valor para nuestros accionistas y compartiendo la responsabilidad por nuestro mundo.

2.2.2 Visión

- Ser la empresa de distribución de bienes tangibles más admirada.
- La esencia de nuestro esfuerzo: "entregar más".
- Nuestra visión refleja fuertemente lo que somos: ambiciosos, optimistas, orientados hacia los resultados.
- Comunica nuestra confianza en que siempre podemos superar los obstáculos y lograr nuestros objetivos.
- Esta característica especial es la que nos hace destacarnos dentro de la multitud.

2.2.3 Servicios

TNT cuenta con métodos de distribución y opciones de ruta, flexibles a requerimientos y necesidades:

- **Servicios de entrega express**

  Servicio de entrega internacional en 48 y 72 horas en la mayoría de los destinos europeos, ciudades principales de EEUU y Latinoamérica, además entrega expresa en el resto del mundo. El servicio incluye entrega puerta a puerta de lunes a viernes en horario de oficina para envíos de documentos y paquetes.

- **Importación express**

  Este servicio ofrece la posibilidad de importar cualquier envío desde más de 168 países de todo el mundo y cancelar en destino con su tarifa local de importación (import express) a través de un proceso simple. Al enviar su solicitud de retiro vía e-mail a nuestro departamento de

atención a cliente o comercial, obtendrá el mismo servicio y características de una exportación, con tiempos expresos predefinidos desde el retiro, puerta a puerta y en horarios de oficina.

- **Carga Ligera Economy Express**

Pensado para envíos de carga liviana o ligera, tanto exportación como importación para pesos mayores a 30 kgs. Tiene como cualidad que en costos es mucho más económico y tiempo de tránsito mayor, el mismo que puede variar desde 3 a 6 días adicionales dependiendo el lugar de origen, pero con la misma calidad de servicio. Incluye entrega puerta a puerta, de lunes a viernes, en horario de oficina hasta 500 kilos. Servicio para paquetes no mayores de 70 kilos o paletizados hasta 500 kilos.

- **Servicio Doméstico (Expreso Diurno, Expreso Nocturno, Priority)**

Este servicio fue desarrollado para solucionar problemas de distribución urgente dentro de la región metropolitana, para envíos de documentos, paquetes y cargas a todas las capitales de Bolivia. El servicio incluye entrega puerta a puerta de lunes a viernes en horario de oficina para envíos de documentos y paquetes. El servicio priority da prioridad de entrega de la correspondencia en destino a primera hora del día o el primero en ser entregado.

2.2.4 Courier

Se denomina Courier a una persona o empresa que se dedica a entregar mensajes, paquetes y correo con correspondencia y documentos, cuando tienen carácter de urgente. También se aplica el término Courier al envío mismo. La distinción entre “servicios postales básicos” y “servicios urgentes” radica en la rapidez del envío y en las prestaciones ofrecidas: una mayor rapidez y seguridad en la recogida, distribución y entrega de los envíos, y en algunas de las siguientes modalidades: recogida en el punto de origen, entrega en mano al destinatario, prueba de entrega, seguimiento y localización de los envíos, lo que implica un mayor precio en relación el correo básico que el usuario está dispuesto a pagar si recibe algunos de esos beneficios.

El término técnico de esta operación es “courier back”. En la aduana local los trámites se concretan en un documento especial, llamado “manifiesto” o “guía” que permite esta clase de envíos a través de agentes, autorizado IATA, llamados “transitorios” **(RAIZ EMPRENDEDOR, 2010)**.

2.2.5 Courier Nacional

El servicio de Courier nacional es el envío de documentos o muestras a distintos puntos al interior de un país, el transporte depende del tipo de envío y medios que existe a la ruta de destino, aéreo o terrestre.

2.2.6 Proceso del servicio de Courier Nacional

Todo envío de correspondencia de documentos o no documentos comprende una serie de procesos:

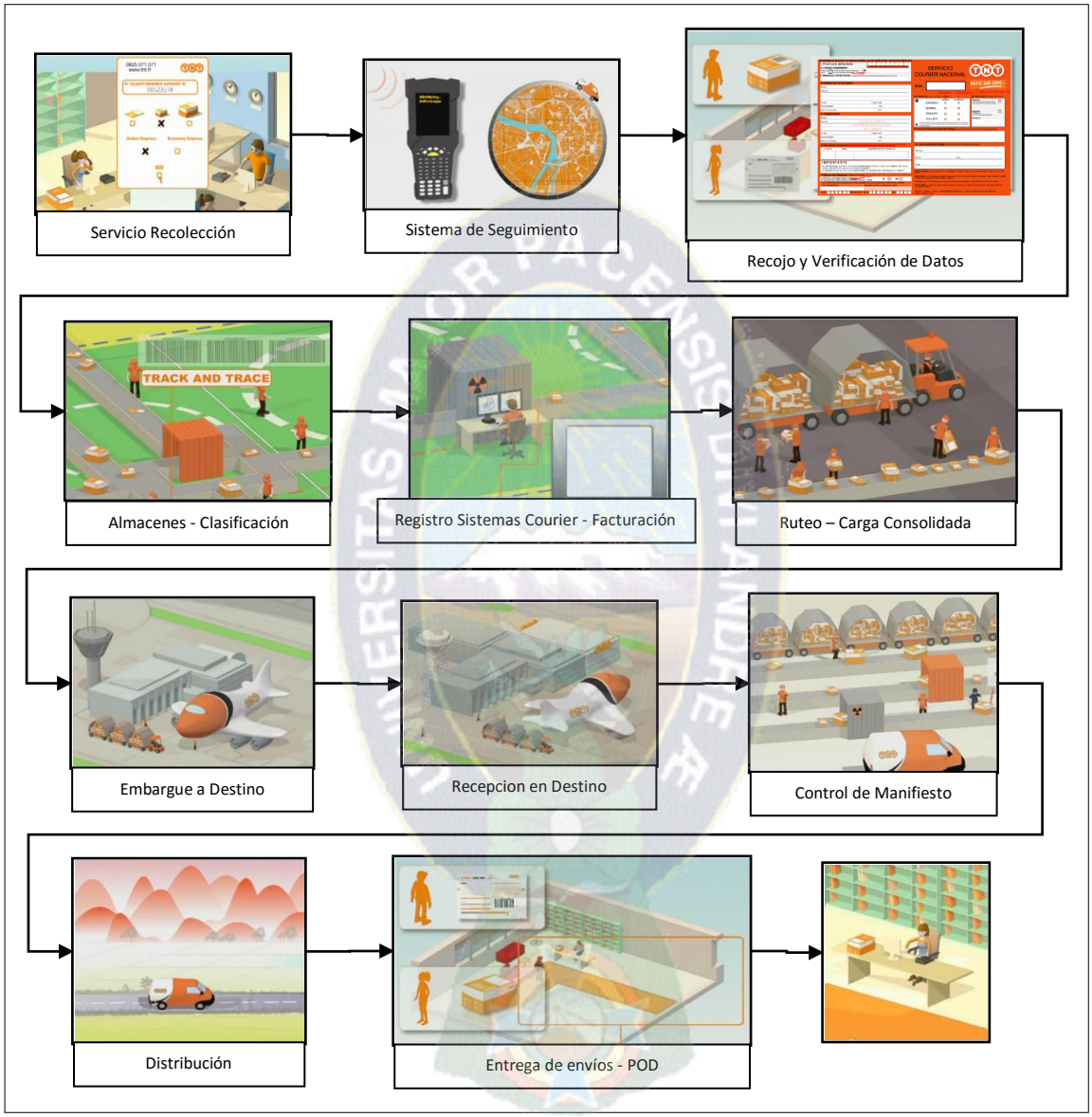
- Coordinación de recojo con el remitente (cliente o empresa) y asignación de hora de recojo.
- Recepción del paquete previa verificación de datos llenados en la guia.
- Zonificación de todas las guias por destino y consolidación de la carga para su despacho.
- Registro de los datos de todas las guias en sistemas internos.
- Envío, transporte y manipuleo de la carga.
- Enviar pre-alerta a destinos para que la carga sea esperada en destino.
- Recepción, verificación y cruce de guias de la carga consolidada en destino.
- Zonificación de todas las guias por zonas en almacenes.
- Entrega al consignatario en dirección señalada y llenado de datos de entrega conforme al cliente.
- Registro y actualización de datos de la entrega.

2.2.7 Definición de Logística

La logística es un mercado diferente de los servicios básicos y de los urgentes y que se diferencia de ambos por la inclusión de nuevas prestaciones de mayor valor añadido. La logística se define como *"una función operativa importante que comprende todas las actividades necesarias para la obtención y administración de materias primas y componentes, así como el manejo de los productos terminados, su empaque y su distribución a los clientes"* **(Introducción A Los Negocios En Un Mundo Cambiante, 2004)**.

En la figura 2.1 se muestra el proceso de la logística que tiene la empresa desde la solicitud de recolección por parte del cliente hasta su entrega en destino.

Figura 2.1: Logística en TNT Express



Fuente: [Elaboración Propia]

2.2.8 Guías

El termino guía es utilizado en este marco del servicio de Courier, al ser definido como el documento físico de los datos llenados por el remitente los cuales sirven para la identificación, clasificación y logística de entrega del envío que viene adjunto, este documento consta de 3 copias y un original:

1 **Original,** queda para el cliente como respaldo que realizo su envío.
2 **1ra. Copia**, sirve para el procesamiento de datos y/o facturación en origen.
3 **2da. Copia**, conformidad que se queda en oficina de destino.
4 **3ra. Copia**, conformidad de entrega para el destinatario y cliente final.

Los datos que ingresa el remitente, Courier y el destinatario están descritos de la siguiente forma:

Tabla 2.1: Estructura de una Guia

NOMBRE – CAMPO	DESCRIPCION – DATOS
Numero de guía	este número es único en su lote entregado al cliente para la identificación, llenado, seguimiento y rastreo.
Datos del remitente o recogida	(nombre, dirección, ciudad, persona de contacto, teléfono, firma del remitente, fecha y hora).
Datos del destinatario	(nombre, dirección, ciudad, provincia, persona de contacto y teléfono).
Descripción del envío	(número de piezas, peso bruto, peso volumen ydescripción de contenido).
Datos del courier	(nombre, firma, fecha y hora).
Tipo de servicio	este varía de acuerdo al tipo de mercancía que se envía y de acuerdo a la urgencia que solicita el cliente (expreso normal, expreso diurno, expreso nocturno, priority o pago en destino).
Instrucciones especiales	este dato puede ser utilizado para especificar el manejo de la mercancía (frágil, interno, entregar a una hora específica, etc.).
Conformidad de entrega	este espacio es para los datos del destinatario, cuando el paquete fue entregado (nombre, fecha, hora y firma).

2.2.9 Recolección

Este servicio es definido como la coordinación de un recojo y envío de una guía elaborada por un cliente, donde el departamento de operaciones coordina que courier o mensajero atenderá esta solicitud y recogerá el paquete de la dirección del cliente.

El proceso de recolección sigue los siguientes pasos:

- Cliente se comunica con TNT e informa que tiene un envío que quiere realizar.
- Servicio al cliente o Cusser (Customer Service) registra todos los datos del cliente y la hora en el cual se puede pasar a recoger el envío y dicta el código que el sistema genera automáticamente asignado.
- El dispatcher es la persona que verifica en el sistema los recojos pendientes y asigna un courier o mensajero que pertenece a la zona para atender el recojo a la hora que el cliente menciono.
- El courier o mensajero una vez recoge el sobre o paquete de la dirección del cliente comunica al dispatcher la hora y cuantos paquetes son para terminar con el servicio de recolección del cliente.

2.2.10 Entregas

- Se consolida toda la carga en almacenes para confirmar la recepción y actualizar los envios recibidos en destino.
- Se clasifica, se rutea (zonifica) y se asigna las guias a cada Courier por zona que les corresponde se despacha todos los envios que saldrán a reparto, todo este proceso es manual
- Se actualiza en el sistema las guias que salieron a entrega.
- El Courier o mensajero al final del día entrega las pruebas de entrega físicas al dataentry (encargado de actualizar las guias), algunas son firmadas otras con únicamente con sello de recepción.
- Los envios no entregados en el día son almacenadas para salir a entrega al día siguiente previa coordinación con el cliente.

2.2.11 Prueba de entrega

Una prueba de entrega es un documento físico donde el destinatario puede firmar o sellar y colocar la hora y fecha que está siendo recibida el envío o su paquete. Esta prueba de entrega es una constancia para la empresa que cumplió con la entrega y fue recibida por parte del cliente.

Este documento puede ser solicitado por el cliente para comprobar y tener respaldo del mismo para fines de cancelación de facturas por el servicio o para presentar como constancia de prueba de recepción.

2.2.12 Manifiestos

Es un documento simplificado que contiene la individualización de las guías courier que se transporta a destino con la información resumida e importante, mediante el cual las encomiendas que llegan a almacenes en destino ayuda para el control, registro, actualización y clasificación. El manifiesto es un resumen de toda la carga que se está enviando y recibiendo cada departamento debe consolidar el mismo y dar parte en qué estado llego la carga, si hubo sobrantes, faltantes, envíos en mal estado, etc.

Toda la información va indexada al número de guía que tiene el paquete la cual refleja en qué estado se encuentra el envío individualmente.

2.2.13 Rastreo y Seguimiento

El rastreo es mantener actualizado la información del estado que se encuentra una guía en particular, es de responsabilidad tanto de origen y destino mantener la información al día para que el cliente y la empresa estén informados de cómo va la situación de su envío,

El seguimiento a una guía es el control que se tiene en todo momento por parte del departamento de operaciones en origen y destino para el cumplimiento y finalizar el proceso de entrega al remitente.

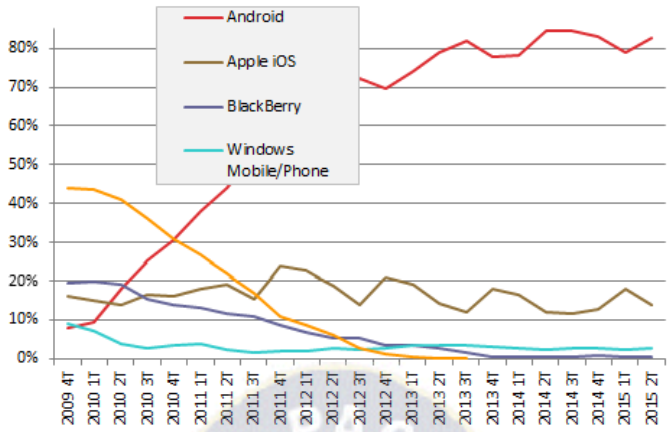
2.3 Tecnologías de desarrollo

2.3.1 Sistemas operativos móviles

Un sistema operativo móvil o so móvil es un sistema operativo que controla un dispositivo móvil como teléfonos Smartphone, tabletas y los últimos relojes inteligentes. Al igual que del pc que utilizan Windows o Linux entre otros, sin embargo, los sistemas operativos móviles son mucho más simples y están más orientados la conectividad inalámbrica, formatos multimedia para móviles y las diferentes maneras de introducir información en ellos.

Algunos sistemas operativos móviles son: google Android, Apple iOS y Windows Phone.

Figura 2.2: Porcentaje de teléfonos inteligentes vendidos según su sistema operativos hasta el segundo semestre de 2015 en el mundo



Fuente: [Gartner Group]

Tabla 2.2: Comparativa de las principales plataformas móviles

	Apple iOS 8	Android 6.0	Windows Phone 8	BlackBerry 10	Firefox OS 2.2
Compañía	Apple	Open Handset Alliance	Microsoft	BlackBerry	Mozilla Foundation
Núcleo del SO	Mac OS X	Linux	Windows NT	QNX	Linux
Licencia de *software*	Propietaria	Libre y abierto	Propietaria	Propietaria	Libre y abierto
Año de lanzamiento	2007	2008	2010	1999	2013
Fabricante único	Sí	No	No	Sí	No
Variedad de dispositivos	Modelo único	Muy alta	Media	Baja	Muy baja
Soporte memoria externa	No	Sí	Sí	Sí	Sí
Motor del navegador web	WebKit	WebKit/ Chromium (Blink)	Trident	WebKit	WebKit
Tienda de aplicaciones	App Store	Google Play	Windows Marketplace	BlackBerry World	Firefox Marketplace
Número de aplicaciones	800.000 (marzo 2013)	800.000 (marzo 2013)	130.000 (enero 2013)	100.000 (enero 2013)	¿?
Coste publicar	$99 / año	$25 una vez	$99 / año	Sin coste	Sin coste
Otras tiendas sin supervisión	No	Si	No	Si	Si
Familia CPU soportada	ARM	ARM, MIPS, x86	ARM	ARM	ARM, x86
Soporte 64 bits	Si	Si	No	No	No
Máquina virtual	No	Dalvik / ART	.net	No	Navegador Web
Lenguaje de programación	Objective-C, C++	Java, C++	C#, Visual Basic, C++	C, C++, Java	HTML5, CSS, JavaScript
Plataforma de desarrollo	Mac	Windows, Mac, Linux	Windows	Windows, Mac	Windows, Mac, Linux
Multiusuario	No	Si	No	No	No
Modo invitado	Si	Si	No	No	No

Fuente: [Gartner Group]

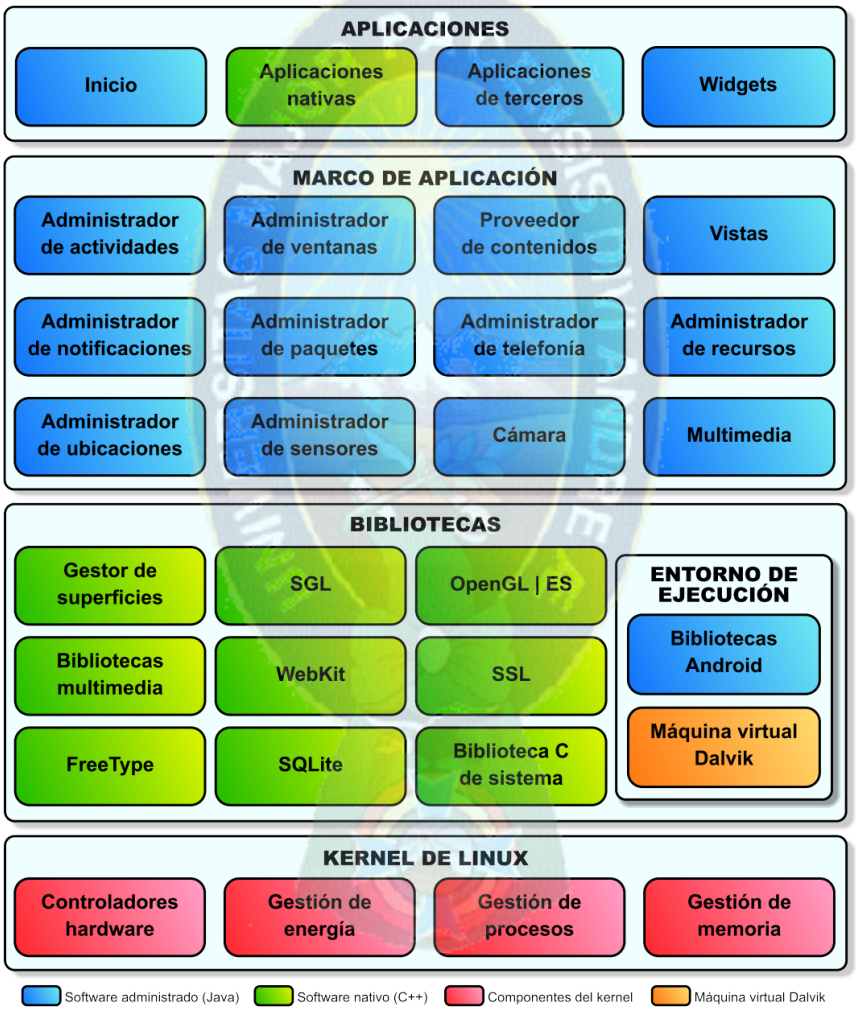
2.3.1.1 Sistema operativo móvil Android

Android es un sistema operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Windows Phone, Symbian, Blackberry os. Lo que lo hace diferente es que está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma.

El sistema permite programar aplicaciones en una variación de java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) De una forma muy sencilla en un lenguaje de programación muy conocido como es java.

2.3.1.2 Arquitectura de Android

Figura 2.3: Arquitectura de Android



Fuente: [Google Inc]

- **Aplicaciones**: este nivel contiene, tanto las incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las api y librerías de los niveles anteriores.

- **Framework de aplicaciones**: representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de api y el mismo "Framework", representado por este nivel.
- **Librerías**: la siguiente capa se corresponde con las librerías utilizadas por Android. Éstas han sido escritas utilizando c/c++ y proporcionan a Android la mayor parte de sus capacidades más características. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android.
- **Tiempo de ejecución de Android:** al mismo nivel que las librerías de Android se sitúa el entorno de ejecución. Éste lo constituyen las core libraries, que son librerias con multitud de clases java y la máquina visual Dalvik.
- **Núcleo Linux**: Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware pueda ser utilizado mediante las llamadas correspondientes. Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerias de control o drivers necesarios dentro de este Kernel de Linux embebido en el propio Android.

2.3.2 Web API

Un api es una interfaz de programación de aplicaciones (del inglés API: Application Programming Interface). Es un conjunto de rutinas que provee acceso a funciones de un determinado software, la salida puede ser, por ejemplo. Json / XML, de entrada, puede ser XML / json / o datos sin formato.

Son publicadas por los constructores de software para permitir acceso a características de bajo nivel o propietarias, detallando solamente la forma en que cada rutina debe ser llevada a cabo y la funcionalidad que brinda, sin otorgar información acerca de cómo se lleva a cabo la tarea. Son utilizadas por los programadores para construir sus aplicaciones sin necesidad de volver a programar funciones ya hechas por otros, reutilizando código que se sabe que está probado y que funciona correctamente.

2.3.3 Código QR

QR (del inglés Quick Response code, “Código de respuesta rápida”). Es un sistema para almacenar información en una matriz de puntos o un código de barras bidimensional creado por la compañía japonesa denso wave, subsidiaria de Toyota, en 1994. Se caracteriza por los tres cuadrados que se encuentran en las esquinas y que permiten detectar la posición del código al lector.

2.3.4 Arquitectura Cliente – Servidor

Es una relación entre procesos corriendo en máquinas separadas, que interactuaran mediante solicitudes y respuestas.

- **Cliente:** realiza requerimiento de servicio al servidor a través de las redes LAN o WAN.
- **Servidor web:** software que se ejecuta en un computador continuamente, a la espera de peticiones de ejecución que realiza el cliente a través de internet. El servidor web aloja los sitios de internet, base de datos con un software instalado el cual gestiona todas las peticiones.

2.3.5 Programación En Tres Capas

Es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.

Capas y niveles

1. **Capa de presentación:** es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato). También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario. Esta capa se comunica únicamente con la capa de negocio.
2. **Capa de negocio:** es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse. Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base

de datos almacenar o recuperar datos de él. También se consideran aquí los programas de aplicación.

3. **Capa de datos:** es donde residen los datos y es la encargada de acceder a los mismos. Está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

2.3.6 PHP

Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas web dinámicas. Se usa principalmente para la interpretación del lado del servidor que se incorpora en el documento HTML. Es considerado unos de los lenguajes más flexibles, potentes y de alto rendimiento. Este lenguaje forma parte del software libre publicado bajo la licencia PHP.

2.3.7 MySQL

Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario. Es muy utilizado en aplicaciones web, como Joomla, Wordpress, Drupal o phpBB, en plataformas (Linux / Windows - Apache – MySQL).

2.3.8 Ajax

Asynchronous JavaScript and XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications). Estas aplicaciones se ejecutan en el cliente, es decir, en el navegador de los usuarios mientras se mantiene la comunicación asíncrona con el servidor en segundo plano. De esta forma es posible realizar cambios sobre las páginas sin necesidad de recargarlas, lo que significa aumentar la interactividad, velocidad y usabilidad en las aplicaciones.

2.3.9 Android Studio

Android Studio es un entorno de desarrollo integrado (IDE) para la plataforma Android. Fue anunciado por Ellie Powers el 16 de mayo de 2013. Android Studio está disponible para desarrolladores para probarlo gratuitamente. Basado en Intellij idea de Jetbrains, está diseñado específicamente para desarrollar para Android. Está disponible para descargar para Windows, Mac xOS y Linux.

2.3.10 JDK Java

Java Development Kit es un conjunto de herramientas (programas y librerías) que permiten desarrollar (compilar, ejecutar, generar documentación, etc.) Programas en lenguaje java.

2.3.11 Microsoft Visio

Es un software de dibujo vectorial para Microsoft Windows. Las herramientas que lo componen permiten realizar diagramas de oficinas, diagramas de bases de datos, diagramas de flujo de programas, UML, y más, que permiten iniciar al usuario en los lenguajes de programación.

Aunque originalmente apuntaba a ser una aplicación para dibujo técnico para el campo de ingeniería y arquitectura; con añadidos para desarrollar diagramas de negocios, su adquisición por Microsoft implicó drásticos cambios de directrices de tal forma que a partir de la versión de visio para Microsoft office 2003 el desarrollo de diagramas para negocios pasó de añadido a ser el núcleo central de negocio, minimizando las funciones para desarrollo de planos de ingeniería y arquitectura que se habían mantenido como principales hasta antes de la compra.

2.3.12 Adobe Dreamweaver

Es una aplicación en forma de suite (basada en la forma de estudio de adobe flash) que está destinada a la construcción, diseño y edición de sitios y aplicaciones web basados en estándares. Es el programa más utilizado en el sector del diseño y la programación web, por sus funcionalidades, su integración con otras herramientas como adobe flash y, recientemente, por su soporte de los estándares del World Wide Web consortium.

2.4 Teoría de colas

Es el estudio matemático del comportamiento de líneas de espera dentro de un sistema. En el caso concreto de la ingeniería, la teoría de colas permite modelas sistemas en los que varios agentes que demandan cierto servicio o prestación de un mismo servidor y, por lo tanto, pueden registrarse esperas desde que un agente llega al sistema y el servidor atiende sus demandas.

En el contexto de la informática y de las tecnologías de la información y la computación las situaciones de espera dentro de una red son más frecuentes, así, por ejemplo, los procesos enviados a un servidor para su ejecución forman colas de espera mientras no son atendidos; la información solicitada, a través de internet a un servidor web puede recibirse con demora debido a la congestión en la red.

Ahora en el contexto de colas de esta presente, cuando los "clientes" llegan a un "lugar" demandando un servicio a un "servidor", el cual tiene una cierta capacidad de atención. Si el

servidor no está disponible inmediatamente y el cliente decide esperar, entonces se forma la línea de espera.

2.4.1 Colas

Una cola Queue en inglés es una estructura de datos caracterizada por ser una lista ordenada de elementos en donde siempre se insertan nuevos elementos al final de la lista y se extraen elementos desde el inicio de la lista, también se conoce como listas FIFO (First In – First Out) el primero en entrar es el primero en salir.

2.4.2 TDA (Tipo de Dato Abstracto)

Un tipo de dato abstracto (TDA) es un conjunto de datos u objetos al cual se le asocian operaciones, el TDA es un modelo matemático que provee de un interfaz con la cual es posible realizar operaciones permitidas, abstrayéndose de la manera en como estén implementadas dichas operaciones.

2.4.3 TDA Cola de Prioridad

Una cola de prioridad es un tipo de datos abstracto que almacena un conjunto de datos que poseen una llave perteneciente a algún conjunto ordenado, y permite insertar nuevos elementos y extraer el máximo (o el mínimo, en caso de que la estructura se organice con un criterio de orden inverso).

Es frecuente interpretar los valores de las llaves como prioridades, con lo cual la estructura permite insertar elementos de prioridad cualquiera, y extraer el de mejor prioridad.

Dos formas simples de implementar colas de prioridad son:

Tabla 2.3: Formas de implementar una cola de prioridad

Lista ordenada	**Lista desordenada**
- Inserción: O(n) - Extracción de máximo: O(1)	- Inserción: O(1) - Extracción de máximo: O(n)

Fuente: [Elaboración Propia]

2.4.4 Heaps

Un heap es un árbol binario de una forma especial, que permite su almacenamiento en un arreglo sin usar punteros. Un heap tiene todos sus niveles llenos, excepto posiblemente el de más abajo, y en este último los nodos están lo más a la izquierda posible

Una cola es una línea de espera y la teoría de colas es una colección de modelos matemáticos que describen sistemas de línea de espera particulares o sistemas de colas. Los modelos sirven para encontrar un buen compromiso entre costes del sistema y los tiempos promedio de la línea de espera para un sistema dado.

2.5 Metodologías agiles

Las metodologías ágiles son una serie de técnicas para la gestión de proyectos que han surgido como contraposición a los métodos clásicos de gestión como CMMI - integración de modelos de madurez de capacidades o Capability Maturity Model Integration aunque surgieron en el ámbito del desarrollo de software, también han sido exportadas a otro tipo de proyectos.

Todas las metodologías que se consideran ágiles cumplen con el manifiesto ágil que no es más que una serie de principios que se agrupan en 4 valores:

- **Los individuos y su interacción**, por encima de los procesos y las herramientas.
- **El software que funciona**, frente a la documentación exhaustiva.
- **La colaboración con el cliente**, por encima de la negociación contractual.
- **La respuesta al cambio**, por encima del seguimiento de un plan.

Se pretende aumentar la eficiencia de las personas involucradas en el proyecto y, como resultado de ello, minimizar el coste. Una de las claves que un proyecto ágil es el uso de un ciclo de vida iterativo, a veces también incremental, frente al ciclo de vida en cascada.

Entre las metodologías ágiles más usadas se encuentran:

- **Scrum**. Es un marco de trabajo que nos proporciona una serie de herramientas y roles para, de una forma iterativa, poder ver el progreso y los resultados de un proyecto.
- **Kanban**. Se basa en una idea muy simple. Ésta es que el trabajo en curso (work in progress, wip) debería limitarse y sólo deberíamos empezar con algo nuevo cuando un bloque de trabajo anterior haya sido entregado o ha pasado a otra función posterior de la cadena.
- **XP**: es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo,

preocupándose por el aprendizaje de los desarrolladores y propiciando un buen clima de trabajo.

2.5.1 Metodologías agiles para el desarrollo de software móvil

El desarrollo de aplicaciones móviles sufre prácticamente los mismos problemas que la gran mayoría de desarrollos de software. Aunque hay que tener en cuenta sus principales peculiaridades como la corta duración de sus desarrollos, la gran competencia del sector que obliga a una constante innovación, los cambios frecuentes en la plataforma de desarrollo y en el hardware o la simplicidad de algunas aplicaciones. Todo ello influye a la hora de elegir una metodología concreta de desarrollo.

2.5.2 Características y Requerimientos Específicos Del Entorno Móvil

El desarrollo de aplicaciones móviles difiere del desarrollo de software tradicional en muchos aspectos, lo que provoca que las metodologías usadas para estos entornos también difieran de las del software clásico. Esto es porque el software móvil tiene que satisfacer una serie de requerimientos y condicionantes especiales que lo hace más complejo:

- **Canal Radio**: consideraciones tales como la disponibilidad, las desconexiones, la variabilidad del ancho de banda, la heterogeneidad de redes o los riesgos de seguridad han de tenerse especialmente en cuenta en este entorno de comunicaciones móviles.
- **Movilidad**: aquí influyen consideraciones como la migración de direcciones, alta latencia debido a cambio de estación base o la gestión de la información dependiente de localización. Sobre esta última, de hecho, se pueden implementar un sinfín de aplicaciones, pero la información de contexto asociada resulta muchas veces incompleta y varía frecuentemente.
- **Portabilidad**: la característica portabilidad de los dispositivos terminales implica una serie de limitaciones físicas directamente relacionadas con el factor de forma de los mismos, como el tamaño de las pantallas (algo que ha variado sustancialmente con la popularización de las pantallas táctiles), o del teclado, limitando también el número de teclas y su disposición.
- **Fragmentación de la industria**: la existencia de una considerable variedad de estándares, protocolos y tecnologías de red diferentes añaden complejidad al escenario del desarrollo móvil.

- **Capacidades limitadas de los terminales**: aquí incluimos factores como la baja potencia de cálculo o gráfica, los riesgos en la integridad de datos, las interfaces de usuario poco funcionales en muchos aspectos, la baja capacidad de almacenamiento, la duración de las baterías o la dificultad para el uso de periféricos en movilidad.

  Factores todos que, por otro lado, están evolucionando en la dirección de la convergencia de las ultras portátiles (netbooks) con los dispositivos inteligentes (Smartphone) constituyendo cada vez menos un elemento diferencial.
- **Diseño**: desde el punto de vista del desarrollo, el diseño multitarea y la interrupción de tareas es clave para el éxito de las aplicaciones de escritorio; pero la oportunidad y frecuencia de éstas es mucho mayor que en el software tradicional, debido al entorno móvil que manejan, complicándose todavía más debido a la limitación de estos dispositivos.
- **Usabilidad**: las necesidades específicas de amplios y variados grupos de usuarios, combinados con la diversidad de plataformas tecnológicas y dispositivos, hacen que el diseño para todos se convierta en un requisito que genera una complejidad creciente difícil de acotar.
- **Time-to-market**: en un sector con un dinamismo propio, dentro de una industria en pleno cambio, los requisitos que se imponen en términos de tiempo de lanzamiento son muy estrictos y añaden no poca dificultad en la gestión de los procesos de desarrollo.

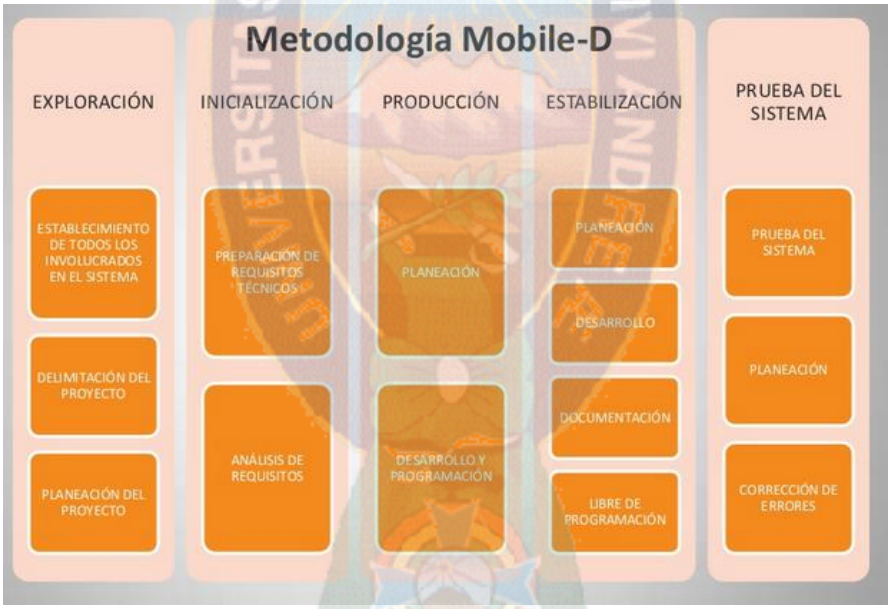
El diseño de sistemas de software móvil es, por tanto, bastante más complejo que el tradicional visto en los otros proyectos de desarrollo, forzando a los investigadores a reconsiderar el uso de las metodologías actuales de desarrollo de software. Como se ha visto, el uso de metodologías ágiles es el medio más apropiado para el desarrollo de tecnología en móviles, aunque las características especiales de los terminales y de las redes de telefonía móvil demandan algunos ajustes sobre las actuales metodologías ágiles.

2.6 Mobile – D

El objetivo de este método es conseguir ciclos de desarrollo muy rápidos en equipos muy pequeños. Fue creado en un proyecto finlandés en 2005, pero sigue estando vigente. Basado en metodologías conocidas pero aplicadas de forma estricta como: eXtreme Programming - XP, Crystal methodologies y Rational Unified Process - RUP. Mobile-D es la metodología de los investigadores del VTT (instituto de investigación finlandés) para el desarrollo de software ágil. Además del desarrollo de software para móviles, es conveniente para los varios contextos, por ejemplo, seguridad, finanzas, logística y aplicaciones de simulación de productos.

Se compone de distintas fases: exploración, inicialización, fase de producto, fase de estabilización y la fase de pruebas. Cada una tiene un día de planificación y otro de entrega.

Figura 2.4: Fases de Mobile-D



Fuente: [UNAD, Realidad Aumentada]

a. **Exploración** se centra la atención en la planificación y a los conceptos básicos del proyecto. Aquí es donde hacemos una definición del alcance del proyecto y su establecimiento con las funcionalidades donde queremos llegar.

b. **Iniciación** configuramos el proyecto identificando y preparando todos los recursos necesarios como hemos comentado anteriormente en esta fase la dedicaremos un día a la planificación y el resto al trabajo y publicación.

c. **Producto** se repiten iterativamente las subfases. Se usa el desarrollo dirigido por pruebas (TDD), antes de iniciar el desarrollo de una funcionalidad debe existir una prueba que verifique su funcionamiento. En esta fase podemos decir que se lleva a acabo toda la implementación.

d. **Estabilización** en la que se realizan las acciones de integración para enganchar los posibles módulos separados en una única aplicación.

e. **Pruebas**. Una vez parado totalmente el desarrollo se pasa una fase de testeo hasta llegar a una versión estable según lo establecido en las primeras fases por el cliente. Si es necesario se reparan los errores, pero no se desarrolla nada nuevo.

Una vez acabada todas las fases deberíamos tener una aplicación publicable y entregable al cliente.

2.7 Programación extrema XP

Extreme Programming (XP) es un enfoque de la ingeniería de software formulado por Kent Beck, el cual busca dos objetivos claramente: hacer un software con calidad y de la forma más rápida posible.

Es el más destacado de los procesos ágiles de desarrollo de software o ágil. La programación extrema se diferencia de las metodologías tradicionales principalmente porque pone más énfasis en la adaptabilidad que en la previsibilidad. **(Extreme Programming Explained: Embrace change, 1999)**.

Los principios básicos se basan en 12 agrupados en 4 categorías.

1. **Retroalimentación a escala fina**
    a. **Principio de pruebas.** Se establece un periodo de pruebas de aceptación del programa (periodo de caja negra), donde se definen las entradas al sistema y los resultados esperados.
    b. **Proceso de planificación.** El usuario escribe sus necesidades para luego crear un documento llamado “User-Stories”. Con esto se define los tiempos de entrega para recibir el feedback por parte del usuario.

c. **El cliente en el sitio.** Se le da poder al cliente para poder determinar los requerimientos, definir la funcionalidad, señalar las prioridades y responder las preguntas de los programadores.

d. **Programación en pareja.** Se supone que la mayor calidad del código escrito de esta manera -el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata.

2. **Proceso continuo en lugar de lotes**

   a. **Integración continua.** Permite al equipo hacer un rápido progreso implementando las nuevas características del software uniendo y reconstruyendo el sistema varias veces al día.

   b. **Refactorización.** Los programadores evalúan continuamente el diseño y re codifican lo necesario, minimizando el código duplicado e ineficiente.

   c. **Entregas pequeñas.** Se coloca un sistema sencillo en producción rápidamente que se actualiza de forma rápida y constante, avaluado en ambiente real.

3. **Entendimiento compartido**

   a. **Diseño simple.** Se basa en la filosofía de que el mayor valor de negocio es entregado por el programa más sencillo que cumpla los requerimientos.

   b. **Metáfora.** Define una historia de cómo funciona el sistema completo.

   c. **Propiedad colectiva del código.** Nadie es propietario de nada todos son el propietario de todo.

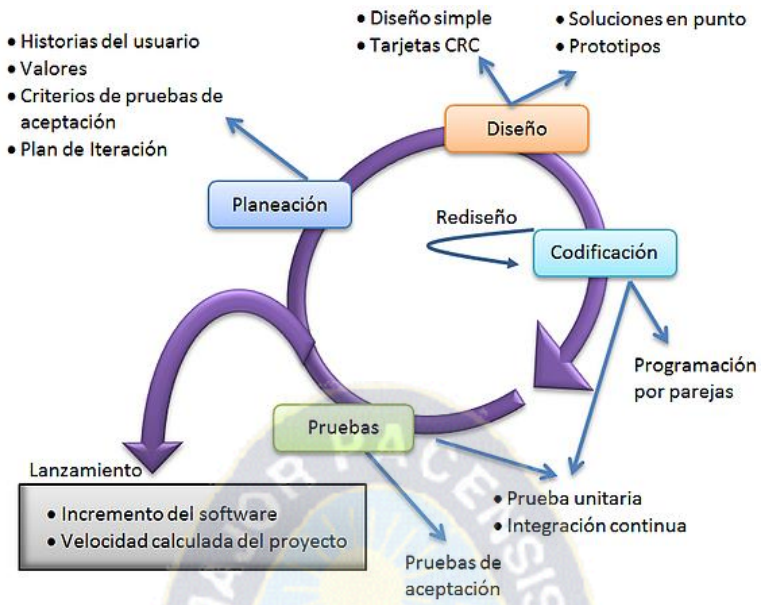
   d. **Estándar de codificación.** Reglas para escribir y documentar el código para que los programadores cumplan y así el sistema pareciera que fue creado por una sola persona.

4. **Bienestar del programador**

   a. **La semana de 40 horas.** Mantener a los programadores frescos, minimizar las horas extras.

La figura 2.5 muestra como las reglas de la programación extrema trabajan con los clientes donde son parte fundamental para el proceso de desarrollo; los desarrolladores contribuyen activamente con su trabajo y los administradores se concentran en la comunicación y las relaciones.

Figura 2.5: Proceso XP



Fuente: [Raquel Murillo M, 2015]

2.8 Valores en la programación extrema (XP)

Los valores son una descripción de cómo debe enfocarse el desarrollo del software:

2.8.1 Simplicidad

La simplicidad es la base de la programación extrema. Se simplifica el diseño para agilizar el desarrollo y facilitar el mantenimiento. Para mantener la simplicidad es necesaria la refactorización del código, ésta es la manera de mantener el código simple a medida que crece. También se aplica la simplicidad en la documentación, de esta manera el código debe comentarse en su justa medida, intentando eso sí que el código esté autodocumentado. Para ello se deben elegir adecuadamente los nombres de las variables, métodos y clases. Aplicando la simplicidad junto con la autoría colectiva del código y la programación por parejas se asegura que cuanto más grande se haga el proyecto, todo el equipo conocerá más y mejor el sistema completo.

2.8.2 Comunicación

Para los programadores el código comunica mejor cuanto más simple sea. El código autodocumentado es más fiable que los comentarios ya que éstos últimos pronto quedan desfasados con el código a medida que es modificado. Debe comentarse sólo aquello que no va a variar, por ejemplo, el objetivo de una clase o la funcionalidad de un método. Las pruebas unitarias son otra forma de comunicación ya que describen el diseño de las clases y los métodos al mostrar ejemplos

concretos de cómo utilizar su funcionalidad. La comunicación con el cliente es fluida ya que el cliente forma parte del equipo de desarrollo. El cliente decide qué características tienen prioridad y siempre debe estar disponible para solucionar dudas.

2.8.3 Retroalimentación (FeedBack)

Al estar el cliente integrado en el proyecto, su opinión sobre el estado del proyecto se conoce en tiempo real. Al realizarse ciclos muy cortos tras los cuales se muestran resultados, se minimiza el tener que rehacer partes que no cumplen con los requisitos y ayuda a los programadores a centrarse en lo que es más importante. Considérense los problemas que derivan de tener ciclos muy largos. El código también es una fuente de retroalimentación gracias a las herramientas de desarrollo. Por ejemplo, las pruebas unitarias informan sobre el estado de salud del código. Ejecutar las pruebas unitarias frecuentemente permite descubrir fallos debidos a cambios recientes en el código.

2.8.4 Coraje o valentía

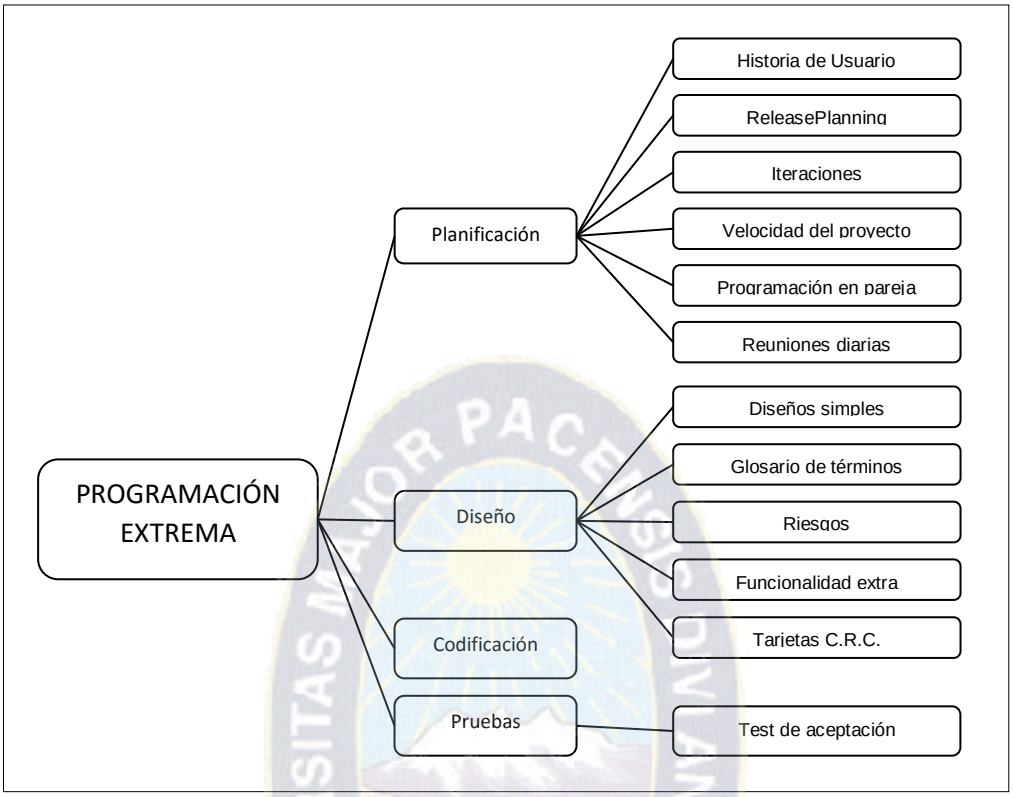
Diseñar y programar para hoy y no para mañana. Esto es un esfuerzo para evitar empantanarse en el diseño y requerir demasiado tiempo y trabajo para implementar todo lo demás del proyecto. La valentía le permite a los desarrolladores que se sientan cómodos con reconstruir su código cuando sea necesario. Saber cuándo desechar un código: valentía para remover código fuente obsoleto, sin importar cuanto esfuerzo y tiempo se invirtió en crear ese código.

2.8.5 Respeto

Los miembros respetan su trabajo porque siempre están luchando por la alta calidad en el producto y buscando el diseño óptimo o más eficiente para la solución a través de la refactorización del código. Los miembros del equipo respetan el trabajo del resto no haciendo menos a otros, sino orientándolos a realizarlo mejor, obteniendo como resultado una mejor autoestima en el equipo y elevando el ritmo de producción en el equipo.

2.9 Fases programación extrema

Figura 2.6: Fases de la Programación Extrema



Fuente: [Elaboración propia]

2.9.1 Planificación

1ª fase: planificación del proyecto

a) Historias de usuario

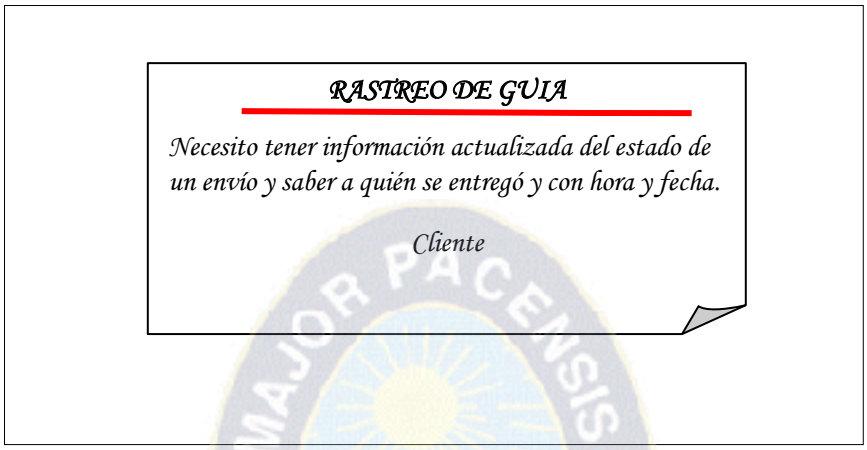
El primer paso de cualquier proyecto que siga la metodología xp es definir las historias de usuario con el cliente. Son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. También se utilizan en la fase de pruebas, para verificar si el programa cumple con lo que especifica la historia de usuario.

El concepto de las User-Stories tiene algo que ver los use cases ("casos de uso") utilizados en el ciclo incremental de desarrollo de software. Nos permiten sustituir unos largos requerimientos por una serie de User-Stories y además nos permiten hacernos una estimar el tiempo para la reunión de lanzamientos de las futuras versiones de nuestro sistema.

Las "User-Stories" son la forma en la cual el usuario entrega los requerimientos del sistema al equipo de trabajo, "La descripción del sistema desde el punto de vista del usuario". Para la

realización de las “User-Stories”, necesitaremos tarjetas de cartulina sin líneas en ellas, de un tamaño determinado, usualmente 4x6 o 5x8.

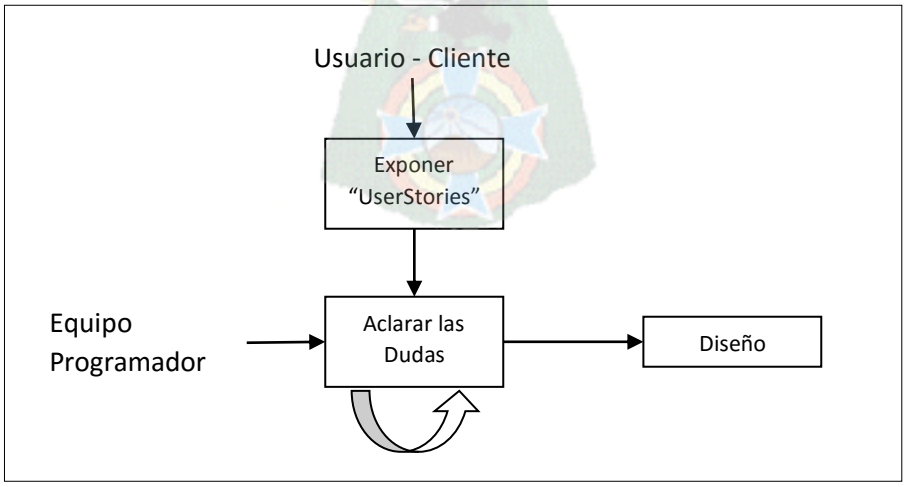
Figura 2.7: Historia de usuarios



Fuente: [Elaboración propia]

Las User-Stories nos ayudan a crear test de aceptación. Estos son pruebas que se aplicarán al sistema para ver si cumplen una determinada “Historia de usuario”, lo cual viene a ser lo mismo que cumplir un determinado requisito en otros modelos de desarrollo. El análisis en la programación extrema es una labor común a los miembros integrantes del equipo de desarrollo y al propio cliente, de esta manera el cliente enuncia sus necesidades y el programador estima los costos y los tiempos.

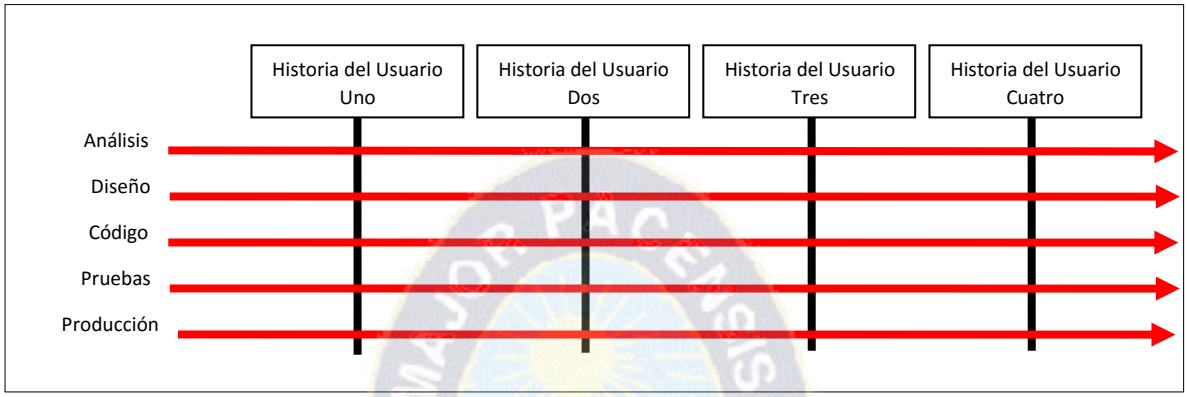
Figura 2.8: Proceso de análisis en programación extrema



Fuente: [Extreme Programming, 2003]

Todos los procesos de análisis, diseño, código, pruebas y producción que se realiza en el proyecto son constantes tomando en cuenta las "User-Stories" para que se consiga lo que el cliente requiere cada vez que lo necesitan.

Figura 2.9: Proceso de Desarrollo con "User-Stories"



Fuente: [eXtreme Programming, 2003]

b) Release planning (plan de publicaciones)

Después de tener ya definidas las historias de usuario es necesario crear un plan de publicaciones, en inglés "Release Plan", donde se indiquen las historias de usuario que se crearán para cada versión del programa y las fechas en las que se publicarán estas versiones. Un "Release Plan" es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa. Después de un "Release Plan" tienen que estar claros estos cuatro factores: los objetivos que se deben cumplir (que son principalmente las historias que se deben desarrollar en cada versión), el tiempo que tardarán en desarrollarse y publicarse las versiones del programa, el número de personas que trabajarán en el desarrollo y cómo se evaluará la calidad del trabajo realizado.

c) Iteraciones

Todo proyecto que siga la metodología XP se ha de dividir en iteraciones de aproximadamente 3 semanas de duración. Al comienzo de cada iteración los clientes deben seleccionar las historias de usuario definidas en el "Reléase Planning" que serán implementadas. También se seleccionan las historias de usuario que no pasaron el test de aceptación que se realizó al terminar la iteración anterior.

d) Velocidad del proyecto

La velocidad del proyecto es una medida que representa la rapidez con la que se desarrolla el proyecto; estimarla es muy sencillo, basta con contar el número de historias de usuario que se pueden implementar en una iteración; de esta forma, se sabrá el cupo de historias que se pueden desarrollar en las distintas iteraciones. Usando la velocidad del proyecto controlaremos que todas las tareas se puedan desarrollar en el tiempo del que dispone la iteración.

e) Programación en pareja

La metodología XP aconseja la programación en parejas pues incrementa la productividad y la calidad del software desarrollado. El trabajo en pareja involucra a dos programadores trabajando en el mismo equipo; mientras uno codifica haciendo hincapié en la calidad de la función o método que está implementando, el otro analiza si ese método o función es adecuado y está bien diseñado. De esta forma se consigue un código y diseño con gran calidad.

f) Reuniones diarias

Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta. Las reuniones tienen que ser fluidas y todo el mundo tiene que tener voz y voto **(eXtreme Programming, 2003).**

2.10 Diseño del sistema

2ª fase: diseño.

a) Diseños simples:

La metodología XP sugiere que hay que conseguir diseños simples y sencillos. Hay que procurar hacerlo todo lo menos complicado posible para conseguir un diseño fácilmente entendible e implementable que a la larga costará menos tiempo y esfuerzo desarrollar.

b) Glosarios de términos:

Usar glosarios de términos y una correcta especificación de los nombres de métodos y clases ayudará a comprender el diseño y facilitará sus posteriores ampliaciones y la reutilización del código.

c) Riesgos

Si surgen problemas potenciales durante el diseño, x.p sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema.

d) Funcionalidad extra

Nunca se debe añadir funcionalidad extra al programa, aunque se piense que en un futuro será utilizada. Sólo el 10% de la misma es utilizada, lo que implica que el desarrollo de funcionalidad extra es un desperdicio de tiempo y recursos.

e) Refactorizar

Refactorizar es mejorar y modificar la estructura y codificación de códigos ya creados sin alterar su funcionalidad. Refactorizar supone revisar de nuevo estos códigos para procurar optimizar su funcionamiento. Es muy común rehusar códigos ya creados que contienen funcionalidades que no serán usadas y diseños obsoletos. Esto es un error porque puede generar código completamente inestable y muy mal diseñado; por este motivo, es necesario refactorizar cuando se va a utilizar código ya creado.

f) Tarjetas C.R.C.

El uso de las tarjetas C.R.C (Class, Responsabilities and Collaboration) permiten al programador centrarse y apreciar el desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedural clásica.

Las tarjetas C.R.C representan objetos; la clase a la que pertenece el objeto se puede escribir en la parte de arriba de la tarjeta, en una columna a la izquierda se pueden escribir las responsabilidades u objetivos que debe cumplir el objeto y a la derecha, las clases que colaboran con cada responsabilidad. **(eXtreme Programming, 2003)**

2.11 Codificación y Desarrollo

3ª fase: Codificación y Desarrollo

Como ya se dijo en la introducción, el cliente es una parte más del equipo de desarrollo; su presencia es indispensable en las distintas fases de XP. a la hora de codificar una historia de usuario su presencia es aún más necesaria. No olvidemos que los clientes son los que crean las historias de usuario y negocian los tiempos en los que serán implementadas. Antes del desarrollo de cada historia de usuario el cliente debe especificar detalladamente lo que ésta hará y también tendrá que estar presente cuando se realicen los test que verifiquen que la historia implementada cumple la funcionalidad especificada.

La codificación debe hacerse ateniendo a estándares de codificación ya creados. Programar bajo estándares mantiene el código consistente y facilita su comprensión y escalabilidad.

Crear test que prueben el funcionamiento de los distintos códigos implementados nos ayudará a desarrollar dicho código. Crear estos test antes nos ayuda a saber qué es exactamente lo que tiene que hacer el código a implementar y sabremos que una vez implementado pasará dichos test sin problemas ya que dicho código ha sido diseñado para ese fin. Se puede dividir la funcionalidad que debe cumplir una tarea a programar en pequeñas unidades, de esta forma se crearán primero los test para cada unidad y a continuación se desarrollará dicha unidad, así poco a poco conseguiremos un desarrollo que cumpla todos los requisitos especificados.

XP sugiere un modelo de trabajo usando repositorios de código dónde las parejas de programadores publican cada poca hora sus códigos implementados y corregidos junto a los test que deben pasar. De esta forma el resto de programadores que necesiten códigos ajenos trabajarán siempre con las últimas versiones. Para mantener un código consistente, publicar un código en un repositorio es una acción exclusiva para cada pareja de programadores.

XP también propone un modelo de desarrollo colectivo en el que todos los programadores están implicados en todas las tareas; cualquiera puede modificar o ampliar una clase o método de otro programador si es necesario y subirla al repositorio de código. El permitir al resto de los programadores modificar códigos que no son suyos no supone ningún riesgo ya que para que un código pueda ser publicado en el repositorio tiene que pasar los test de funcionamiento definidos para el mismo.

La optimización del código siempre se debe dejar para el final. Hay que hacer que funcione y que sea correcto, más tarde se puede optimizar.

A la hora de codificar no seguimos la regla de XP que aconseja crear test de funcionamiento con entornos de desarrollo antes de programar. Nuestros test los obtendremos de la especificación de requisitos ya que en ella se especifican las pruebas que deben pasar las distintas funcionalidades del programa, procurando codificar pensando en las pruebas que debe pasar cada funcionalidad. **(eXtreme Programming, 2003)**

2.12 Pruebas

4ª fase: Pruebas.

Uno de los pilares de la metodología XP es el uso de test para comprobar el funcionamiento de los códigos que vayamos implementando.

a) El uso de los test en XP

Se deben crear las aplicaciones que realizarán los test con un entorno de desarrollo específico para test. Hay que someter a test las distintas clases del sistema omitiendo los métodos más triviales. Se deben crear los test que pasarán los códigos antes de implementarlos; en el apartado anterior se explicó la importancia de crear antes los test que el código.

Un punto importante es crear test que no tengan ninguna dependencia del código que en un futuro evaluará. Hay que crear los test abstrayéndose del futuro código, de esta forma aseguraremos la independencia del test respecto al código que evalúa.

Como se comentó anteriormente los distintos test se deben subir al repositorio de código acompañados del código que verifican. Ningún código puede ser publicado en el repositorio sin que haya pasado su test de funcionamiento, de esta forma, aseguramos el uso colectivo del código (explicado en el apartado anterior).

El uso de los test es adecuado para observar la refactorización. Los test permiten verificar que un cambio en la estructura de un código no tiene por qué cambiar su funcionamiento.

b) Test de aceptación

Los test mencionados anteriormente sirven para evaluar las distintas tareas en las que ha sido dividida una historia de usuario. Para asegurar el funcionamiento final de una determinada historia de usuario se deben crear "test de aceptación"; estos test son creados y usados por los clientes para comprobar que las distintas historias de usuario cumplen su cometido.

Al ser las distintas funcionalidades de nuestra aplicación no demasiado extensas, no se harán test que analicen partes de las mismas, sino que las pruebas se realizarán para las funcionalidades generales que debe cumplir el programa especificado en la descripción de requisitos

2.13 Modelos de negocio

Algo muy importante para un proyecto de software es que esté correctamente alineado con el modelo de negocio para el que ha sido creado.

El problema surge debido a que los modelos de negocio cambian cada vez más rápido, a que las empresas deben actuar cada vez contra más elementos y más variados (respuestas ante nuevas tecnologías, adecuar los procesos de negocio a estas, razones de competitividad, etc.). Nuestro sistema de desarrollo de software debe ser capaz de responder de una manera lo suficientemente rápida a los cambios en las necesidades de nuestro cliente.**: (eXtreme Programming, 2003).**

2.14 Leguaje Unificado de Modelado - UML

UML es un lenguaje de propósito general para el modelado orientado a objetos, que combina notaciones provenientes desde: Modelado Orientado a Objetos, Modelado de Datos, Modelado de Componentes, Modelado de Flujos de Trabajo (Workflows).

Para este proyecto utilizaremos UML para poder interpretar en la fase de Diseño del sistema y complementar la metodología Mobile-D, modelando el sistema utilizando los Casos de uso.

2.14.1 Casos de Uso

Los Diagramas de Caso de Uso, son diagramas que describen el comportamiento del sistema cuando algo o alguien (actor) interactúan con el sistema en relación con el negocio. Este comportamiento se puede explicar de forma gráfica y/o textual, describiendo la naturaleza del estímulo que proyecta el caso de uso. Esto conlleva a puntualizar, que ningún sistema se encuentra aislado.

Los Diagramas de Caso de Uso han proveído un medio para que los desarrolladores, los usuarios finales del sistema y los expertos del dominio lleguemos a una comprensión común del sistema. Perteneciendo a la segunda fase del Rational Unified Process (RUP)

2.15 Métricas de Calidad de Software

La calidad de un sistema web es el conjunto de características y atributos de calidad de un sistema que satisface os requerimientos de calidad establecidos o implicados.

2.15.1 Modelo de Mc Call

La **ISO 9126** basada en el modelo de Mc Call plantea un modelo normalizado que permite evaluar y comparar productos sobre la misma base.

Aquí la calidad queda definida a un alto nivel de abstracción por seis características:

- **Funcionalidad**: las funciones satisfacen necesidades declaradas o implícitas **(ISO 9126: 1991)**
- **Fiabilidad:** capacidad de un sistema para mantener su nivel de rendimiento
- **Usabilidad:** esfuerzo necesario para el uso y la valoración individual de tal uso, por parte de un conjunto de usuarios. **(ISO 9126: 1991)**
- **Portabilidad:** es la capacidad de un sistema para ser transferido de un entorno a otro. **(ISO 9126: 1991)**

- **Mantenibilidad:** es el esfuerzo necesario para realizar modificaciones específicas. **(ISO 9126: 1991)**
- **Eficiencia:** es la relación entre el nivel de prestaciones de un sistema y el volumen de recursos utilizados en condiciones declaradas. **(ISO 9126: 1991)**

Este estándar no proporciona métricas ni métodos de medición, por lo que no son prácticas las mediciones directas de las características de calidad.

Para resolver este problema se revisó la **ISO 9126** y se incluyó un nuevo modelo de calidad que distingue entre tres aproximaciones a la calidad de producto en **ISO 14598**:

- **Calidad interna:** se mide por las propiedades estáticas del código, utilizando técnicas de inspección.
- **Calidad externa:** se mide por las propiedades dinámicas del código cuando éste se ejecuta.
- **Calidad en uso:** se mide por el grado por el cual el software está realizado en función de las necesidades del usuario en el entorno de trabajo para el que fue construido.

2.16 COCOMO (COnstructive COst MOdel)

Es un modelo de estimación de costos, esfuerzo y tiempo que refleja los cambios en la práctica de desarrollo de software profesional.

Entre los distintos métodos de estimación de costes de desarrollo de software, el modelo COCOMO (COnstructive COst MOdel) desarrollado por Barry M. Boehm, se engloba en el grupo de los modelos algorítmicos que tratan de establecer una relación matemática la cual permite estimar el esfuerzo y tiempo requerido para desarrollar un producto.

Por un lado, COCOMO define tres modos de desarrollo o tipos de proyectos:

- **Orgánico:** proyectos relativamente sencillos, menores de 50 KDLC líneas de código, en los cuales se tiene experiencia de proyectos similares y se encuentran en entornos estables.
- **Semi-acoplado:** proyectos intermedios en complejidad y tamaño (menores de 300 KDLC), donde la experiencia en este tipo de proyectos es variable, y las restricciones intermedias.
- **Empotrado:** proyectos bastante complejos, en los que apenas se tiene experiencia y se engloban en un entorno de gran innovación técnica. Además, se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

Y por otro lado existen diferentes modelos que define COCOMO:

- **Modelo básico:** Se basa exclusivamente en el tamaño expresado en LDC.
- **Modelo intermedio:** Además del tamaño del programa incluye un conjunto de medidas subjetivas llamadas conductores de costes.
- **Modelo avanzado:** Incluye todo lo del modelo intermedio además del impacto de cada conductor de coste en las distintas fases de desarrollo.

A continuación de definen las siguientes fórmulas:

a) **E = Esfuerzo = a KLDC e * FAE** (persona x mes)
b) **T = Tiempo de duración del desarrollo = c Esfuerzo d** (meses)
c) **P= Personal = E/T** (personas)

Para calcular el Esfuerzo, se necesita hallar la variable KDLC (Kilo-líneas de código), donde los PF son 261,36 (dato conocido) y las líneas por cada PF se detallan en la tabla que se ilustra a continuación:

La fórmula para obtener los KDLC será el siguiente:

- **KLDC= (PF * Líneas de código por cada PF)/1000**

Así pues, en nuestro caso el tipo orgánico será el más apropiado ya que el número de líneas de código no supera los 50 KLDC, y además el proyecto no es muy complejo, por consiguiente, los coeficientes que usaremos serán las siguientes:

Tabla 2.4: Coeficiente para tipos de proyecto

PROYECTO SOFTWARE	a	e	c	d
Orgánico	3,2	1,05	2,5	0,38
Semi-acoplado	3,0	1,12	2,5	0,35
Empotrado	2,8	1,20	2,5	0,32

Por otro lado, también se debe hallar la variable FAE, la cual se obtiene mediante la multiplicación de los valores evaluados en los diferentes 15 conductores de coste que se observan en la siguiente tabla:

Tabla 2.5: Conductores de Costo

CONDUCTORES DE COSTE	VALORACIÓN					
	Muy bajo	***Bajo***	***Nominal***	***Alto***	***Muy alto***	***Extr. alto***
Fiabilidad requerida del software	0,75	0,88	1.00	**1,15**	1,40	-
Tamaño de la base de datos	-	0,94	**1.00**	1,08	1,16	-
Complejidad del producto	0,70	**0,85**	1.00	1,15	1,30	1,65
Restricciones del tiempo de ejecución	-	-	1.00	**1,11**	1,30	1,66
Restricciones del almacenamiento princip	-	-	**1.00**	1,06	1,21	1,56
Volatilidad de la máquina virtual	-	0,87	**1.00**	1,15	1,30	-
Tiempo de respuesta del ordenador	-	0,87	1.00	**1,07**	1,15	-
Capacidad del analista	1,46	1,19	1.00	**0,86**	0,71	-
Experiencia en la aplicación	1,29	1,13	1.00	0,91	**0,82**	-
Capacidad de los programadores	1,42	1,17	1.00	0,86	**0,70**	-
Experiencia en S.O. utilizado	1,21	1,10	**1.00**	0,90	-	-
Experiencia en el lenguaje de programaci	1,14	1,07	1.00	**0,95**	-	-
Prácticas de programación modernas	1,24	1,10	**1.00**	0,91	0,82	-
Utilización de herramientas software	1,24	1,10	1.00	**0,91**	0,83	-
Limitaciones de planificación del proyecto	1,23	**1,08**	1.00	1,04	1,10	-

2.16.1 Justificación de los valores

2.16.2 Atributos de software

- **Fiabilidad requerida del software**: Si se produce un fallo por el pago de un pedido, o fallo en alguna reserva, etc. puede ocasionar grandes pérdidas a la empresa (Valoración Alta).

- **Tamaño de la base de datos:** La base de datos de nuestro producto será de tipo estándar (Valoración Nominal).
- **Complejidad del producto:** La aplicación no va a realizar cálculos complejos (Valoración Baja).

2.16.3 Atributos de hardware

- **Restricciones del tiempo de ejecución:** En los requerimientos se exige alto rendimiento (Valoración Alta).
- **Restricciones del almacenamiento principal:** No hay restricciones al respecto (Valoración Nominal).
- **Volatilidad de la máquina virtual:** Se usarán sistemas de la "Familia Windows" (Valoración Nominal).
- **Tiempo de respuesta del ordenador:** Deberá ser interactivo con el usuario (Valoración Alta).

2.16.4 Atributos del personal

- **Capacidad del analista:** Capacidad alta relativamente, debido a la experiencia en análisis en proyecto similar (Valoración Alta)
- **Experiencia en la aplicación:** Se tiene cierta experiencia en aplicaciones de esta envergadura (Valoración muy alta).
- **Capacidad de los programadores:** Teóricamente deberá tenerse una capacidad muy alta por la experiencia en anteriores proyectos similares (Valoración muy alta).
- **Experiencia en S.O. utilizado:** Con Windows 2000 Professional la experiencia es a nivel usuario (Valoración Nominal).
- **Experiencia en el lenguaje de programación:** Es relativamente alta, dado que se controlan las nociones básicas y las propias del proyecto (Valoración Alta).

2.16.5 Atributos del proyecto

- **Prácticas de programación modernas:** Se usarán prácticas de programación mayormente convencional (Valoración Nominal).
- **Utilización de herramientas software:** Se usarán herramientas estándar que no exigirán apenas formación, de las cuales se tiene cierta experiencia (Valoración Alta).
- **Limitaciones de planificación del proyecto:** Existen pocos límites de planificación. (Valoración Baja).

2.16.6 Cálculo del esfuerzo del desarrollo:

$$E = a\ KLDC^{e} * FAE \Rightarrow \text{Personas /Mes}$$

2.16.7 Cálculo tiempo de desarrollo:

$$T = c\ Esfuerzo^{d} \Rightarrow \text{Meses de Trabajo}$$

2.16.8 Productividad:

$$PR = LDC/E \Rightarrow \text{Lineas de codigo/personas mes}$$

2.16.9 Personal promedio:

$$P = E/T \Rightarrow \text{\# Personas}$$

2.17 Seguridad

La seguridad en la arquitectura de referencia de los servicios web

En esta sección vamos a estudiar los requisitos de seguridad enumerados por la especificación que define la arquitectura de referencia de los servicios web **(W3C, 2004)** y mencionaremos los distintos tipos de ataques, que también señala esta especificación, contra los que se verán expuestas las soluciones diseñadas basadas en servicios web.

2.17.1 Encriptación MD5

En criptografía, MD5 (abreviatura de Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5) es un algoritmo de reducción criptográfico de 128 bits ampliamente usado. es uno de los algoritmos de reducción criptográficos diseñados por el profesor Ronald Rivest del MIT (Massachusetts Institute of Technology, Instituto Tecnológico de Massachusetts). Fue desarrollado en 1991 en remplazo del algoritmo MD4.

2.17.2 Servicios de seguridad básicos

La seguridad es un concepto más de entre aquellos servicios que comprenden la calidad del servicio (QoS) en los servicios web (los otros tres son la transaccionalidad, fiabilidad, y gestión/administración) **(Endrei et al., 2004a).** Los servicios de seguridad básicos mencionados por la ISO 7498-2 son la confidencialidad, integridad, autenticidad de origen, no repudio y control de acceso y serán estos aspectos sobre los que centraremos el caso de estudio desarrollado en el siguiente apartado.

Por su parte, la arquitectura de referencia planteada por el w3c para los servicios web (w3c, 2004) hace mención a que para garantizar la seguridad en los servicios web es necesario “un amplio espectro de mecanismos que solventen problemas como la autenticación, el control de acceso

basado en roles, la aplicación efectiva de políticas de seguridad distribuidas o la seguridad a nivel de los mensajes". Definamos los servicios de seguridad básicos encontrados en cualquier sistema distribuido:

2.17.3 Autenticación de los interlocutores

Cada servicio web participante en una interacción podría requerir autenticación de la otra parte. Cuando cierto servicio a dirige una petición al servicio b, éste puede requerirle unas credenciales junto con una demostración de que le pertenecen como por ejemplo un par nombre de usuario (credencial)/password (demostración) o un certificado x.509v3 (credencial)/firma digital (demostración). El principal problema de la autenticación, cuyo origen radica en la naturaleza heterogénea de los servicios web, es conseguir un acuerdo en los protocolos y en los formatos de los datos de seguridad empleados. Otro punto a resolver es definir un modelo de autenticación single Sign-On de forma que un servicio web a que necesita interactuar con otros 6 servicios web, para completar un proceso de negocio p en tiempo real, no necesite autenticarse más que una vez frente al primero de ellos para poder completar la operación bajo un tiempo de respuesta aceptable.

2.17.4 Autorización

Los servicios web deben disponer de mecanismos que les permitan controlar el acceso a sus servicios (recursos). Se debe poder determinar quién y cómo puede hacer a qué y cómo sobre sus recursos. La autorización concede permisos de ejecución de ciertos tipos de operaciones sobre ciertos recursos a ciertas identidades autenticadas.

Figura 2.10: Seguridad punto a punto frente a la seguridad extremo a extremo f

Fuente: [seguridad en servicios web,2001]

CAPITULO 3

3 Diseño del Sistema

La aplicación móvil desarrollada tiene como fin principal controlar la entrega de correspondencia y actualizar el estado que se encuentra cada envío en tiempo real.

La metodología utiliza para el desarrollo de la aplicación móvil es Mobile-D el cual tiene las siguientes fases presentadas en la figura 3.1.

Figura 3.1: Fases de Mobile-D

Fuente: [Elaboración propia]

3.1 Fase de exploración

En esta fase se describen los usuarios, las historias de usuario y las tareas de los usuarios.

3.1.1 Descripción de usuarios

La descripción de los usuarios que participan en el proceso Operativo del servicio de courier se presentan a continuación.

Tabla 3.1: Descripción de usuarios

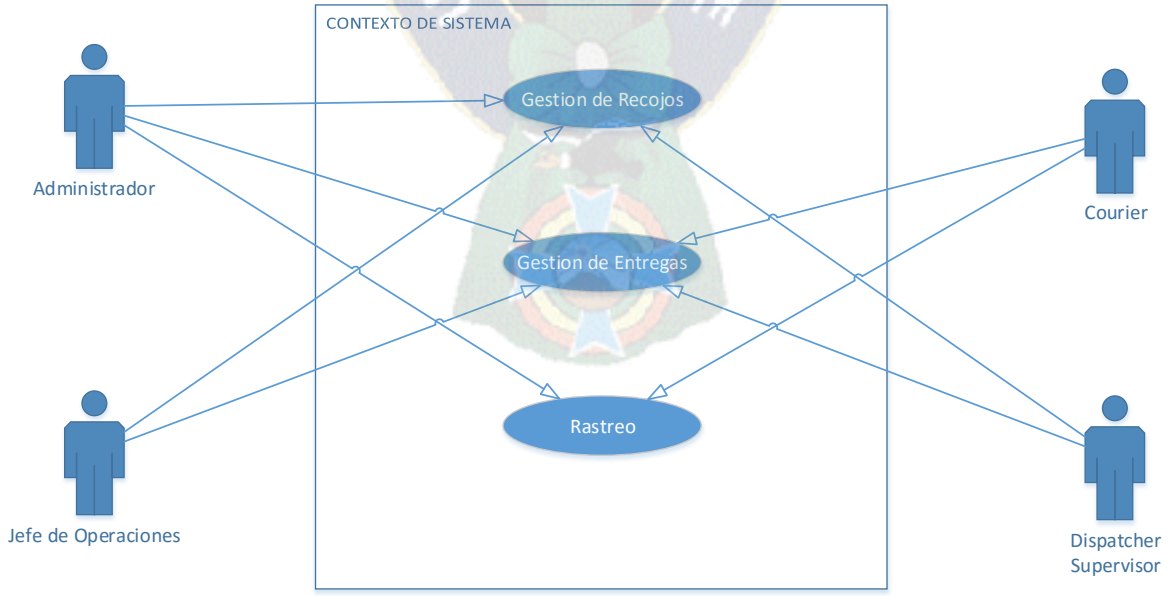
Código	Usuario	Descripción
Us-01	Courier / mensajero	El Courier o mensajero es una persona que realiza los recojos y entregas de toda la correspondencia, las tareas que puede realizar son: consulta de recojos de la fecha actual y entregas pendientes hasta la fecha que le corresponden a su zona. Además, puede realizar consultas de rastreo de guia.
Us-02	Dispatcher	El dispatcher es la persona encargada de controlar y coordinar los recojos y entregas que debe realizar el Courier

		entre sus tareas en el sistema son: Asignar a un courier un recojo programado con un rango de hora estimado. Consultar los recojos y entregas programadas en el sistema
Us-03	Administrador / Jefe de operaciones	El administrador es el encargado de controlar las entregas y recojos además de poder gestionar todo el sistema en caso que se requiera alguna modificación entre sus tareas se tiene:

3.1.2 Rational Unified Process RUP – UML

En este caso para complementar la metodología Mobile-D y para fines de comprensión en esta etapa se apoyará en RUP para el uso del Lenguaje Unificado de Modelado UML por ser una metodología adaptable al contexto y se adecua a necesidades de cada organización, a continuación se presenta el diagrama de Contexto del sistema y Casos de Uso.

3.1.2.1 Diagrama de Contexto



3.1.1.1 Diagrama de Casos de Uso

3.1.3 Requerimientos iniciales

Los requerimientos iniciales identificados son los siguientes:

- Rastreo de guias
- Listado de recojos programados para el día
- Listado de recojos programados asignados por courier o por zona
- Listado de entregas pendientes de entrega o actualización
- Listado de guias según prioridad
- Listado de courier y zonas
- Actualización de recojo programado
- Actualización de guias entregadas
- Actualización de guias recogidas y confirmación de entrega
- Cargar la foto de prueba de entrega al servidor web
- Guardar la prueba de entrega (foto) en dispositivo
- Capturar firma digital del destinatario
- Guardar firma digital en el dispositivo móvil
- Cargar la foto digital al servidor web
- Listado de teléfonos corporativos
- Noticias corporativas

3.1.4 Análisis de requerimientos iniciales

En base al estudio de los casos y las encuestas realizadas se identificó la importancia de cada requerimiento que permitan crear los componentes e interfaces más importantes. La importancia está en una escala de 1 a 10 donde 10 es la mayor importancia.

La consulta es realizada a todo el personal relacionado con el proceso de recolección y entrega de los envios, y se toma el resultado promedio de cada uno de los requerimientos.

Tabla 3.2: Descripción de usuarios

Requerimiento	**Importancia**
✓ Rastreo de guias	6
✓ Listado de recojos programados para el día	9
✓ Listado de recojos programados asignados por courier o por zona	7,2
✓ Listado de entregas pendientes de entrega o actualización	9,5
✓ Listado de guias según prioridad	7
✓ Listado de courier y zonas	6
✓ Actualización de recojo programado	8
✓ Actualización de guias entregadas	9.5
✓ Actualización de guias recogidas y confirmación de entrega	7
✓ Cargar la foto de prueba de entrega al servidor web	9
✓ Guardar la prueba de entrega (foto) en dispositivo	6
✓ Cargar la foto digital al servidor web	9,5
✓ Listado de teléfonos corporativos	4
✓ Noticias corporativas	5

En base a los requerimientos iniciales se ha podido identificar los diferentes módulos listados a continuación.

1. **Módulo de rastreo**
    a. Búsqueda de numero de guia

b. Detalle de guia

2. **Módulo de recojos**
    a. Listado de recojos programados
    b. Detalles de recojo programado
    c. Asignación de courier y zona por parte del dispatcher
    d. Actualización y confirmación de recojo por parte del courier
    e. Captura de datos de las guias recogidas
    f. Captura de coordenadas geo posicionales
3. **Módulo de entregas**
    a. Listado de entregas del día
    b. Listado de entregas pendientes de entrega
    c. Búsqueda de guia pendiente
    d. Actualización de estatus
    e. Actualización y confirmación de entrega por parte del courier
    f. Lectura y actualización de quía por medio de código de barras o código qr
    g. Captura de fotografía de la prueba de entrega
    h. Almacenamiento de fotografía y/o firma digital en el dispositivo móvil y en el servidor web
4. **Modulo noticias**
    a. Listado de noticias corporativas
5. **Modulo mapas**
    a. Visualización de mapa para referencias geo posicionales
6. **Módulo de comunicación**
    a. Llamada directa a soporte de sistemas

3.1.5 Limitaciones

Las limitaciones relacionadas con la plataforma móvil son:

- La aplicación requiere la versión de Android 4.1 o superior
- Es necesario plan de datos móvil

3.1.6 Supuestos y dependencias

- Para el correcto funcionamiento es importante que el usuario tenga conocimientos del manejo de un dispositivo móvil Smartphone en plataforma Android.
- Los datos son extraídos de la base de datos web y actualizados en línea.

- Los usuarios deben estar conscientes del tiempo de vida de la batería y controlar el uso del teléfono móvil.
- Los usuarios deben revisar y estar seguros que los datos son correctos a la hora de confirmar y actualizar cualquier información.

3.2 Fase de Inicialización

3.2.1 Establecimiento del proyecto

En esta etapa se establecen los recursos físicos y técnicos para el proyecto, así también el ambiente para monitorearlo, y se establecen los medios de comunicación con los grupos o usuarios de interés.

3.2.2 Configuración del ambiente de desarrollo

Esta tarea consiste en configurar los ambientes tanto físicos como técnicos para el proyecto. Esta tarea involucra a los desarrolladores de software en el ambiente técnico de desarrollo, además se realizan prueba de concepto sin necesariamente implementar algún requerimiento.

3.2.2.1 Configuración de servidor web

- **Servidor Mysql:**
    - Versión: 5.1.60-community
    - PhpMyAdmin: 3.3.10.1
- **Servidor web:** apache/1.3.41 (unix) mod_fastcgi /2.4.6 mod_auth_pop/ ferozo-2.0 mod_auth_pam/1.1.1 php/4.4.4
    - Versión del cliente: 5.0.94
    - Extensión php: mysql

3.2.2.2 Configuración de App móvil

- **Tipo de proyecto:** Android Studio versión 1.2 preview 2015
- **Configuraciones:** Ninguna especifico
- **Version de android:** Android 4.1 jelly bean
- **Librerías extras:** Barcode Scaner proyecto código abierto Zxing, apache licence 2.0

3.2.3 Entrenamiento

Es una tarea en la cual el equipo de desarrollo se entrena de acuerdo a las necesidades específicas. Incluye procesos de desarrollo, metodología, colección de datos, problemas técnicos relacionados como nuevas herramientas y métodos. En este caso

3.2.4 Planificación inicial

3.2.4.1 Análisis de requerimientos y pre-requisitos

Con el fin de realizar los requerimientos funcionales para el usuario es necesario cumplir con los pre-requisitos a nivel técnico que permitan implementar el requerimiento funcional.

3.2.4.2 Gestión de recojos

Pre-requisitos:

- Registro, consulta, actualización y confirmación de recojos
    - Registro, consulta y actualización de asignación de courier o zona
        - Registro, consulta y actualización de courier / mensajero
        - Registro, consulta y actualización de zonas
        - Registro, consulta y actualización de empresas / clientes

3.2.4.3 Gestión de entregas

Pre-requisitos:

- Registro, consulta, actualización y confirmación de entrega
    - Registro, consulta y actualización de entregas por fecha
        - Listado de guias ordenadas por prioridad
- Registro, almacenamiento y cargado de fotografía de confirmación de entrega
- Registro, almacenamiento y cargado de firma digital de destinatario

3.2.4.4 rastreo de guias

Pre-requisitos:

- Consulta y actualización de guias

3.2.4.5 Noticias corporativas

Pre-requisitos:

- Consulta y actualización de página web

3.2.5 Planificación de fases

Tabla 3.3: Planificación de fases

Fase	Iteracion	Descripción
Exploración		

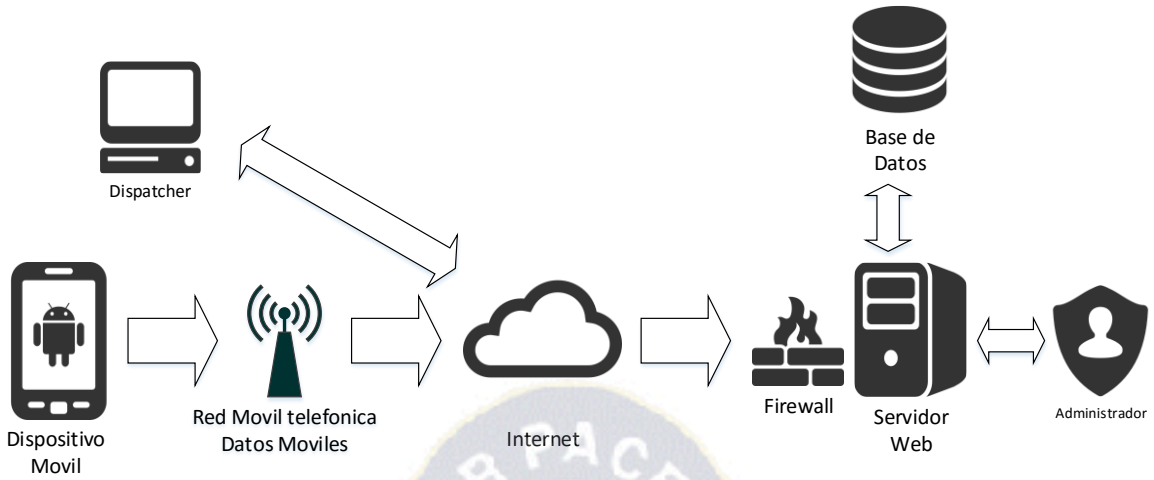
Inicialización	Iteración 0	Establecimiento del proyecto, entrenamiento, análisis de requerimientos iniciales.
Producción	Iteración módulo de recojos	Implementación de módulo de recojos. Refinamiento y actualización de historia de usuario. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de entregas	Implementación de módulo de recojos. Refinamiento y actualización de historia de usuario. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de rastreo	Implementación de módulo de recojos. Refinamiento y actualización de historia de usuario. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de noticias	Implementación de módulo de recojos. Refinamiento y actualización de historia de usuario. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteracion módulo de mapas	Implementación de módulo de recojos. Refinamiento y actualización de historia de usuario. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteracion módulo de corporativos	Implementación de módulo de recojos. Refinamiento y actualización de historia de usuario. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.

Estabilización	Iteración módulo de recojos	Refactorización del módulo de recojos. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de entregas	Refactorización del módulo de recojos. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de rastreo	Refactorización del módulo de recojos. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de noticias	Refactorización del módulo de recojos. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de mapas	Refactorización del módulo de recojos. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración módulo de corporativos	Refactorización del módulo de recojos. Refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
Pruebas del sistema	Iteración pruebas del sistema	Se realiza la evaluación de las pruebas y se realiza el análisis de los resultados.

3.3 Fase de Producción

3.3.1 Diseño del Sistema

Figura 3.2: Diseño general del sistema



Fuente: [Elaboración Propia]

3.3.2 Historia de usuarios

Para estos casos formalizaremos las historias de usuario utilizando la siguiente plantilla.

Tabla 3.4: Plantilla de historia de usuario

Historia de Usuario				
Código				
Nombre				
Actor				
Descripción	"Como (Actor) quiero (funcionalidad) que (beneficio)." (Cohn, 2004)			
HU Relacionada(s):	**Código:**		**Nombre:**	
Módulo				
Criterios de aceptación	**Condición**		**Resultado**	
	Cuando se Acción + Concepto		Se debe cumplir que + ESPECIFICACIÓN	

Las historias de usuario que tiene la aplicación móvil son dos; el registro o actualización de estatus de cada envío o guia (ver HU 001) y consulta de recojos de la fecha actual.

Tabla 3.5: Historia de usuarios: Listado de entregas

Historia de usuario		
Código	001	
Nombre	Listado de entregas	
Actor	Courier / mensajero	
Descripción	Como courier quiero consultar cuales son los envios pendientes de entrega hasta la fecha	
HU relacionadas	002 – administración de entregas	
Modulo	Consulta de entregas	
Criterios de aceptación	**Condición**	**Resultado**
	Cuando se elige botón de listar	Se debe cumplir que el courier este logueado en el sistema y que tenga plan de datos o wifi habilitado.
	Cuando se elige el botón buscar	Se debe cumplir que el número de guia este introducido en el campo de búsqueda

Tabla 3.6: Historia de usuario: Confirmar entrega

Historia de usuario		
Código	002	
Nombre	Confirmar entrega	
Actor	Courier / mensajero	
Descripción	Como courier quiero confirmar la entrega de un envío ingresando el número de guia y eligiendo un estatus de cierre	
HU relacionadas	001 – listado de entregas	
Modulo	Confirmación de entrega de envío	
Criterios de aceptación	**Condición**	**Resultado**

	Cuando se elige el número de guia	Se debe cumplir que el courier este logueado en el sistema y que tenga plan de datos o wifi habilitado.
	Cuando se elige el botón buscar	Se debe cumplir que el número de guia este introducido en el campo de búsqueda
	Cuando se elige el número de guia	Se debe elegir el estatus que corresponde a ese número de guia
	Cuando se elige el número de guia (2)	(2) segunda opción de actualización, para tomar la fotografía a la prueba de entrega o capturar la firma digital del destinatario
	Cuando se elige el botón de confirmar	Se debe cumplir que el estatus y comentarios estén completos.

Tabla 3.7: Historia de usuario: Listado de recojos

Historia de usuario		
Código	003	
Nombre	Listado de recojos	
Actor	Courier / mensajero	
Descripción	Como courier quiero consultar cuales son los recojos programados en la fecha actual	
HU relacionadas	004 – administración de recojos	
Modulo	Consulta de recojos	
Criterios de aceptación	**Condición**	**Resultado**
	Cuando se elige botón de listar	Se debe cumplir que el courier este logueado en el sistema y que tenga plan de datos o wifi

		habilitado.
	Cuando se elige el registro del recojo	Se debe cumplir que el recojo este asignado al courier y completar los datos de numero de guías y demás datos solicitados.
	Cuando se elige el botón de code QR	Se debe leer el código de barras o código QR de la guia y confirmar los datos
	Cuando se elige el botón de entrega confirmada o compañía cerrada	Se debe confirmar el estatus de cierre

Tabla 3.8: Historia de usuario: Confirmar recojos

Historia de usuario		
Código	004	
Nombre	Confirmar recojos	
Actor	Courier / mensajero	
Descripción	Como courier quiero confirmar el recojo de un envío programado y asignado.	
HU relacionadas	001 – listado de recojos	
Modulo	Confirmación de recojo de envío	
Criterios de aceptación	**Condición**	**Resultado**
	Cuando se elige el número de guia	Se debe cumplir que el courier este logueado en el sistema y que tenga plan de datos o wifi habilitado.
	Cuando se elige el registro de recojo	Se debe completar los datos solicitados del recojo
	Cuando se elige el botón confirmar y cargar guías	Se debe completar los datos de los números de guia con el destino.

	Cuando se elige el botón de dar alta guías al servidor	Se confirma que se debe cargar datos al servidor

Tabla 3.9: Historia de usuario: Rastreo de guia

Historia de usuario		
Código	005	
Nombre	Rastreo de guia	
Actor	Courier / mensajero / usuario	
Descripción	Como usuario necesito hacer el rastreo de una guia y me muestre los datos de esa guia con todos sus detalles y el estatus que tenga.	
HU relacionadas		
Modulo	Rastreo de guia	
Criterios de Aceptación	**Condición**	**Resultado**
	Cuando se ingrese al módulo de rastreo de guia	Se debe cumplir que el courier este logueado en el sistema y que tenga plan de datos o wifi habilitado.
	Cuando se elige el botón buscar	Se debe cumplir que el número de guia este introducido en el campo de búsqueda

Tabla 3.10: Historia de usuario: Login e ingreso al sistema

Historia de usuario	
Código	006
Nombre	Logueo de sistema
Actor	Usuarios
Descripción	Como usuario necesito que solo yo pueda ingresar al sistema para poder

	trabajar en la aplicación que requiero	
HU relacionadas		
Modulo	Logueo de usuario en el sistema	
Criterios de aceptación	**Condición**	**Resultado**
	Cuando ingrese al sistema pulsando el icono del sistema	Se debe cumplir que el usuario tenga habilitado plan de datos o wifi.
	Cuando el usuario presione el boton conectarse al sistema	Se debe ingresar los datos correctamente de usuario y contraseña.

3.3.3 Detalle de las Historias de usuario

A continuación de manera resumida todas las historias de usuario y sus tareas correspondientes.

Tabla 3.11: Detalles de historia de usuarios y tareas

Código	**Historia de usuario**	**Tareas**
HC 001	Listado y confirmación de entregas	Diseño y desarrollo del módulo de entregas
HC 002	Listado y confirmación de recojos	Diseño y desarrollo del módulo de recojos
HC 003	Rastreo de guia	Diseño y desarrollo del módulo de rastreo
HC 004	Login de acceso al sistema	Diseño y desarrollo del acceso al sistema
HC 005	Noticias corporativas	Diseño y desarrollo del módulo de noticias
HC 006	Teléfonos corporativos	Diseño y desarrollo del módulo de teléfonos corporativos

3.4 Fase de Estabilización y Fase de Pruebas – Corrección

3.4.1 Tarjeta CRC

Las tarjetas CRC (clase-responsabilidad-colaboración) son una herramienta de BrainStorming, usado como metodología para el diseño de software orientado a objetos creada por Kent Beck y Ward Cunningham.

3.4.2 Casos de Prueba

Un caso de prueba o test case es, en ingeniería del software, un conjunto de condiciones o variables bajo las cuáles un analista determinará si una aplicación, un sistema software (software system), o una característica de éstos es parcial o completamente satisfactoria.

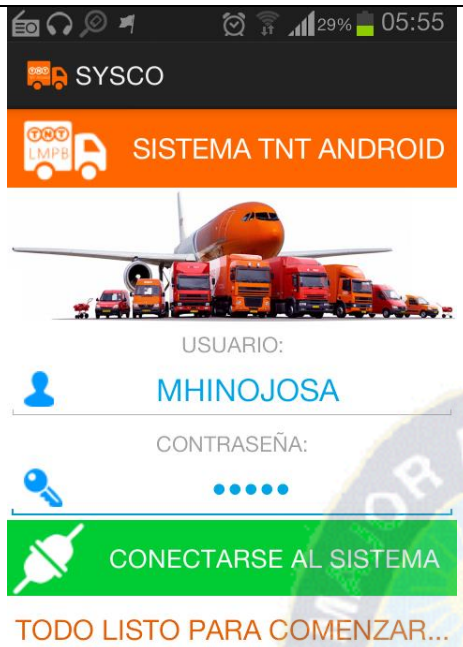
Se pueden realizar muchos casos de prueba para determinar que un requisito es completamente satisfactorio. Con el propósito de comprobar que todos los requisitos de una aplicación son revisados, debe haber al menos un caso de prueba para cada requisito a menos que un requisito tenga requisitos secundarios. En ese caso, cada requisito secundario deberá tener por lo menos un caso de prueba. Algunas metodologías como RUP recomiendan el crear por lo menos dos casos de prueba para cada requisito. Uno de ellos debe realizar la prueba positiva de los requisitos y el otro debe realizar la prueba negativa.

3.4.3 Interfaz de usuario

3.4.3.1 Login del sistema

Tabla 3.12: Login del sistema

Login de sistema

	- El login es el layout principal que se muestra al inicio al ingresar a la aplicación, en las que destaca el botón de conectarse al sistema y todo listo para comenzar cuando el plan de datos móviles o wifi se encuentra habilitado - En caso contrario invita al usuario a activar su plan de datos caso contrario se cierra el sistema

- **Tarjeta CRC**

Login de sistema	
- Permite ingresar al sistema ingresando el usuario y contraseña. - Módulo de seguridad para que solo el usuario registrado pueda ingresar al sistema. - Comprueba si el plan de datos móviles está habilitado.	- Courier - Interfaz de usuario para que pueda ingresar su user y password

- **Casos de Prueba**

Pruebas De Aceptación		
Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
ID de Historia de Usuario	Login del Sistema	
Ensayador	Courier	Courier

Entradas:	User (TextView) = MHINOJOSA Password (TextView) = 4926450	
Resultado Esperado:	Mensaje (MsgBox) = Conectando a Servidor Mensaje (MsgBox) = Bienvenido a Sysco Mensaje (MsgBox) = Error al Iniciar sesión Mensaje (MsgBox) = Su Plan de Datos esta desactivado Mensaje (MsgBox) = ¿No tienes habilitado tus Datos Quieres activarlos?	
Aprobación	Cumple con los controles de Acceso posibles	Cumple con los controles de Acceso posibles
Gravedad	Importante y Critico	Importante y Critico
Reproducibilidad	Único Acceso	Único Acceso

Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
Nombre de UI	Login del Sistema	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios		
Aprobación	Cumple con los controles de Acceso posibles	Cumple con los controles de Acceso posibles
Gravedad	Importante y Critico	Importante y Critico
Reproducibilidad	Layout principal de Inicio	Layout Principal de Inicio

3.4.3.2 Menú principal

Tabla 3.13: Menu principal

Menú principal	
	- El menú principal es el layout principal que permite el ingreso a todos los módulos. - La característica es que muestra el id del usuario su nombre el depot o regionales que pertenece y la zona asignada

- **Tarjeta CRC**

Menú principal	
- Permite ingresar al sistema ingresando el usuario y contraseña. - Módulo de seguridad para que solo el usuario registrado pueda ingresar al sistema. - Comprueba si el plan de datos móviles esta habilitado.	- Courier - Interfaz de usuario para que pueda ingresar su user y password.

- **Casos de Prueba**

Pruebas De Aceptación		
Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
ID de Historia de Usuario	Menu Principal	
Ensayador	Dispatcher	Encargado de Sistemas

Entradas:	Ninguna – Layout de visualization	
Resultado Esperado:	Id Usuario (TextView) Nombre (TextView) Depot (TextView) Zona (TextView) Noticias (Button) Mis Mapas (Button) Rastreo (Button) Mis Recojos (Button) Mis Entregas (Button) POD Droid (Button) Acerca de (Button)	
Aprobación	Cumple con la visualización de los módulos	Cumple con la visualización de los módulos
Gravedad	Importante	Importante
Reproducibilidad	Único Acceso	Único Acceso

Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
Nombre de UI	Menú Principal	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios	Acceso mediante el layout de Login	Acceso mediante el layout de Login
Aprobación	Cumple con los requisitos	Cumple con los requisitos

Gravedad	Importante	Importante
Reproducibilidad	Layout de Visualización	Layout de visualización

3.4.3.3 Modulo noticias

Tabla 3.14: Modulo de noticias

Módulo de noticias	
	- El modulo noticias es el layout donde permite consultar algunas novedades respecto a la empresa, algunas notificaciones, problemas o simplemente asistencia al servicio al clientes. - Este layout es actualizado automáticamente con la página web de la empresa.

- **Tarjeta CRC**

Modulo noticias	
- Permite visualizar noticias que son actualizadas en la página web. - Módulo de lectura.	- Courier - Interfaz de usuario para que pueda visualizar el layout correspondiente

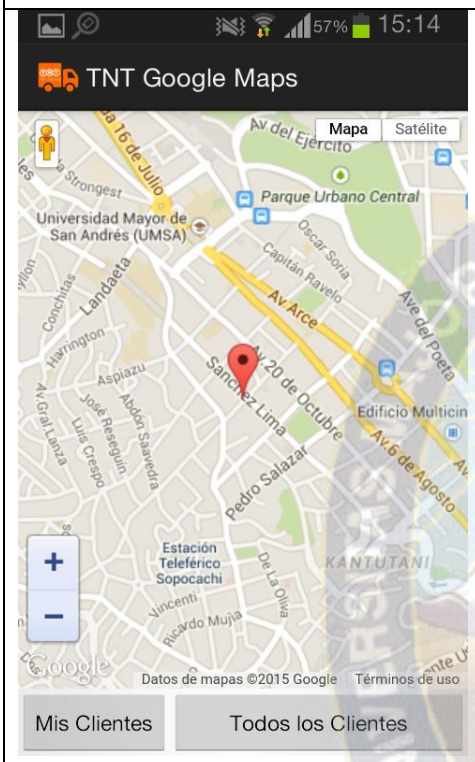
- **Casos de Prueba**

Pruebas De Aceptación		
Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
ID de Historia de Usuario	Noticias TNT	
Ensayador	Dispatcher	Encargado de Sistemas
Entradas:	Ninguna – Layout de visualization	
Resultado Esperado:	Titulo de Noticias (TextView) Campo Web (WebView) = Pagina web service www.tnt.com.bo/noticias	
Aprobación	Cumple con la visualización de los módulos	Cumple con la visualización de los módulos
Gravedad	Normal	Normal
Reproducibilidad	Único Acceso	Único Acceso

Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
Nombre de UI	Noticias TNT	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios	Acceso desde Menu Principal	Acceso desde Menu Principal
Aprobación	Cumple con los requisitos	Cumple con los requisitos
Gravedad	Normal	Normal
Reproducibilidad	Layout de Visualización con Web Service	Layout de visualización con Web Service

3.4.3.4 Modulo mis mapas

Tabla 3.15: Modulo mis mapas

Módulo de mis mapas	
	- Este layout permite visualizar un mapa mostrando la ubicación del dispositivo debe estar conectado a la red móvil de datos o wifi. - Puede mostrarse un mapa satelital o mapa de trazos. - El objetivo es almacenar la posición geográfica del usuario y también mostrar la ubicación de las empresas o clientes con cuenta asignada.

- **Tarjeta CRC**

Modulo mis mapas	
- Permite visualizar un mapa geográfico de la librería de google maps mostrando la ubicación actual el dispositivo	- Courier - Interfaz de usuario para que pueda visualizar un mapa en la posición actual.

- **Casos de Prueba**

Pruebas De Aceptación		
Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4

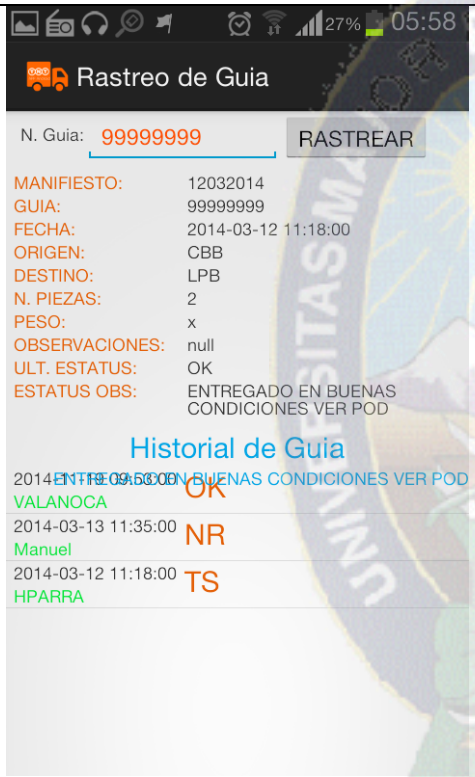
ID de Historia de Usuario	Mis Mapas	
Ensayador	Dispatcher	Encargado de Sistemas
Entradas:	Desplazamiento de Layout Touch de Entrada para punto señalado Button geolozalizacion de coordenadas de los clientes de la zona actual Button geolicalizacion de coordenadas de todos lo clientes	
Resultado Esperado:	Titulo de Mapas (TextView) Campo <fragment> = Maps Service de google maps Mis Clientes (Button) Todos los Clientes (Button)	
Aprobación	Cumple con la visualización del mapa	Cumple con la visualización del mapa
Gravedad	Normal	Normal
Reproducibilidad	Único Acceso	Único Acceso

Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
Nombre de UI	Mis Mapas	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios	Acceso desde Menu Principal	Acceso desde Menu Principal
Aprobación	Cumple con los requisitos	Cumple con los requisitos
Gravedad	Normal	Normal

Reproducibilidad	Layout de Visualización con desplazamiento en Maps Service de Google	Layout de Visualización con desplazamiento en Maps Service de Google

3.4.3.5 Modulo rastreo de guia

Tabla 3.16: Modulo rastreo

Módulo de rastreo	
	- El modulo de rastreo permite hacer el rastreo en línea de una guia digitada para tener datos de su estado o comprobar la entrega confirmada. - Este layout permite obtener datos importantes de la guia desde la fecha enviada hasta el peso y quien la recibió.

- **Tarjeta CRC**

Modulo rastreo	
- Permite buscar un envío según su número de guia.	- Courier - Interfaz de usuario para que pueda ingresar un numero de guia y poder buscar y visualizar los datos principales de esa guia.

- **Casos de Prueba**

Pruebas De Aceptación		
Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
ID de Historia de Usuario	Rastreo de Guias	
Ensayador	Dispatcher	Encargado de Sistemas
Entradas:	Numero de guia (TextView) = Numero de guia a rastrear Rastrea (Button) = Click para rastreo de guia	
Resultado Esperado:	Listado (ListView) = Listado de Estatus de cada guia Detalles (ListView) = Detalles de envio (Numero, Origen, Destino, Peso, Piezas, Observaciones, Tipo de Servicio)	
Aprobación	Cumple con la visualización de los estatus de la guia	Cumple con la visualización de los estatus de la guia
Gravedad	Normal	Normal
Reproducibilidad	Único Acceso	Único Acceso

Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
Nombre de UI	Rastreo de Guia	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios	Acceso desde Menu Principal	Acceso desde Menu Principal
Aprobación	Cumple con los requisitos	Cumple con los requisitos
Gravedad	Normal	Normal

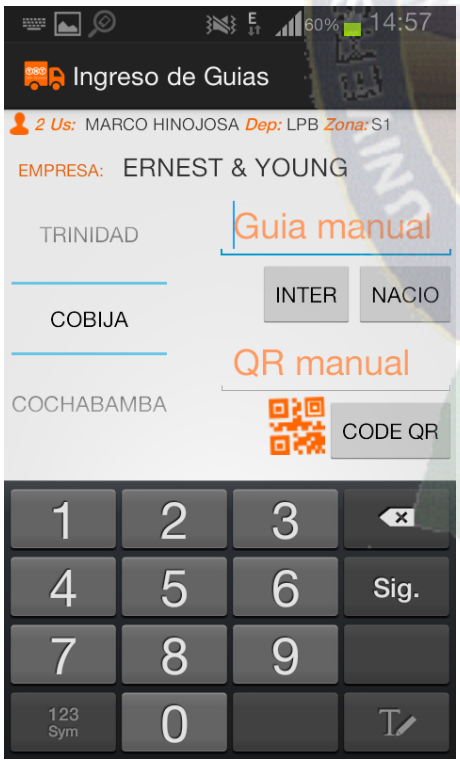
Reproducibilidad	Layout de Visualización	Layout de Visualización

3.4.3.6 Modulo mis recojos

Tabla 3.17: Modulo mis recojos

Módulo de mis recojos	
	- El módulo de recojos permite visualizar todos los recojos programados y asignado a un courier o mensajero. - Permite obtener datos de numero de recojo, dirección hora asignada y algún cobro extra que debe realizarse. - Una vez elegido el recojo se procede a cargar los datos de cantidad de guias nacionales o internaciones que se recogerán. - Cuando se confirma y se carga las guias se procede a ingresar los números de guias ya sea por lectura de código de barras o simplemente ingresando el numero manualmente. - Se debe completar los datos de destino y luego cargar estos al servidor web y actualizar el recojo.

14:56
Detalle de Recojo
2 Us: MARCO HINOJOSA Dep: LPB Zona: S1
CODIGO LLAMADA: 585369
EMPRESA: ERNEST & YOUNG
DIRECCION: AV. 20 DE OCTUBRE # 2665 ED. TORRE AZUL BUILDING P
HORA LLAMADA: 09:00:00
HORA ASIGNADA: 18:45:00
CONTROL: 09:00:00 14:56:52 18:45:00
RESPONSABLE: Sistema
OBSERVACIONES:
Cliente Fijo
CANTIDAD DE GUIAS RECEPCIONADAS
NACIONALES: 00
INTERNACIONALES: 00
CONFIRMAR Y CARGAR GUIAS


14:57
Ingreso de Guias
2 Us: MARCO HINOJOSA Dep: LPB Zona: S1
EMPRESA: ERNEST & YOUNG
TRINIDAD
Guia manual
INTER
NACIO
COBIJA
QR manual
COCHABAMBA
CODE QR
1 2 3
4 5 6 Sig.
7 8 9
123 Sym 0

- **Tarjeta CRC**

Modulo mis recojos	
- Permite gestionar un recojo programado por atención al cliente - Permite actualizar un recojo con datos de guias recogidas. - Permite el registro de número de guias nacionales e internaciones. - Permite elegir el destino de cada envío.	- Courier - Interfaz de usuario para que pueda visualizar y listar datos de recojos y actualizar el mismo.

- **Casos de Prueba**

Pruebas De Aceptación		
Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
ID de Historia de Usuario	Mis Recojos	
Ensayador	Dispatcher	Encargado de Sistemas
Entradas:	Actualizr recojos (Button)	
Resultado Esperado:	Listado (ListView) = Listado de recojos programados (Nombre de Clientes, hora, estatus)	
Aprobación	Cumple con la visualización del listado de recojos	Cumple con la visualización del listado de recojos
Gravedad	Importante Critico	Importante Critico
Reproducibilidad	Único Acceso	Único Acceso

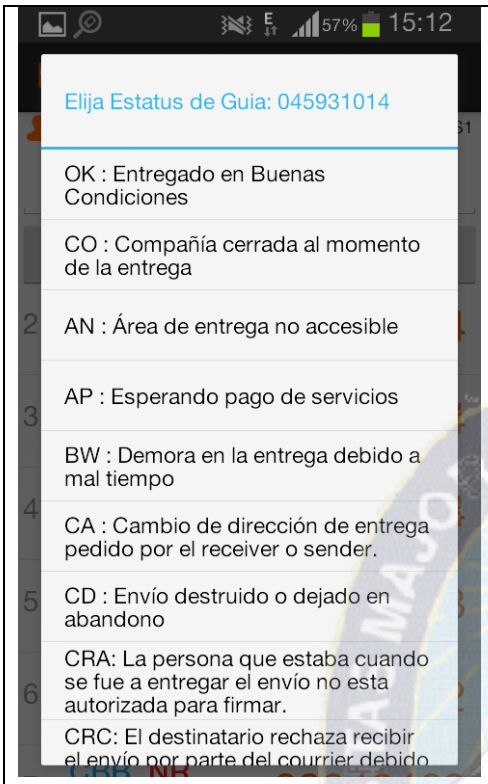
Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play

Versión SW	Android 4.1	Android 4.4
Nombre de UI	Mis recojos	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios	Acceso desde Menu Principal	Acceso desde Menu Principal
Aprobación	Cumple con los requisitos	Cumple con los requisitos
Gravedad	Importante Critico	Importante Critico
Reproducibilidad	Layout de Visualización	Layout de Visualización

3.4.3.7 Modulo mis entregas

Tabla 3.18: Modulo mis entregas

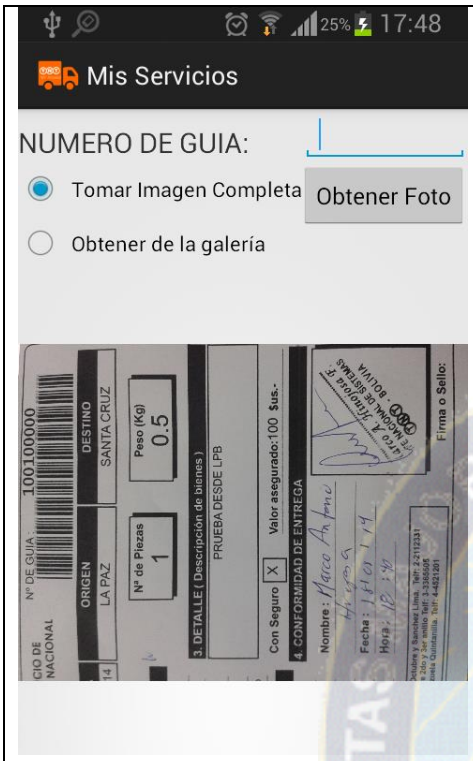
Modulo de mis entregas	
	- Este módulo es el más importante de los mencionados ya que consolida y da fin a la ciclo de vida de un envío. - Permite la búsqueda de una guia o envío para confirmar algún estatus de entrega o simplemente de algún problema con ese envío. - Al elegir una guia se procede a elegir el estatus que corresponde ósea algún comentario de si fue entregado en buenas condiciones o la empresa se encuentra cerrada. - Existen 45 estatus los cuales pueden



ser actualizados a un envío en particular e ingresando algún comentario para ayudar a entender el mismo.

- La otra forma de actualizar una guia es sacándole una foto al comprobante de entrega para subirlo al servidor web para luego ser consultada por el cliente.

- También permite la firma digital para luego ser convertida en imagen y subida al servidor web.

- **Tarjeta CRC**

Modulo mis entregas	
- Permite registrar una entrega. - Permite visualizar y listar los envios pendientes de entrega. - Permite tomar una foto a la prueba de entrega del destinatario. - Permite actualizar con estatus una determinada guia. - Permite registrar la firma digital del destinatario.	- Courier - Interfaz de usuario para que pueda gestionar las entregas pendientes.

- **Casos de Prueba**

Pruebas De Aceptación

Nombre de Teléfono	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
ID de Historia de Usuario	Mis Entregas	
Ensayador	Dispatcher	Encargado de Sistemas
Entradas:	Busqueda de Guia (textView) Actualizar recojos (Button)	
Resultado Esperado:	Listado (ListView) = Listado de guias pendientes de entrega (Numero de Guia, Manifiesto, fecha, Estatus)	
Aprobación	Cumple con la visualización del listado de entregas	Cumple con la visualización del listado de entregas
Gravedad	Importante Critico	Importante Critico
Reproducibilidad	Único Acceso	Único Acceso

Pruebas De Interfaz De Usuario		
Nombre de Modulo	Samsung Galaxy S3 mini	Huawei G Play
Versión SW	Android 4.1	Android 4.4
Nombre de UI	Mis Entregas	
Ensayador	Dispatcher	Encargado de Sistemas
Prueba de criterios	Acceso desde Menu Principal	Acceso desde Menu Principal
Aprobación	Cumple con los requisitos	Cumple con los requisitos
Gravedad	Importante Critico	Importante Critico
Reproducibilidad	Layout de Visualización y Layout de Actualización y toma de foto de prueba de entreg	Layout de Visualización y Layout de Actualización y toma de foto de prueba de entrega

CAPITULO 4

4. Seguridad, Control y Análisis Costo/Beneficio

4.1 Seguridad

Las medidas prácticas de seguridad para las aplicaciones móviles, tales como una buena gestión de dispositivos móviles y el cifrado de datos almacenados, pueden mitigar los riesgos de la entrega de aplicaciones móviles. Así que es imperativo entender los riesgos inherentes asociados con los dispositivos móviles, las medidas de seguridad nativas integras en los sistemas operativos móviles aportan a este objetivo.

Entonces la seguridad parte desde el ingreso y acceso al equipo móvil con un patrón de seguridad, un PIN o una contraseña. A partir de ahí para el ingreso a la aplicación se parte desde la Identificación y Autenticación. Para el caso de la aplicación móvil también como medida de control se realiza el envio y registro de la dirección MAC del dispositivo, conjuntamente en el momento de autenticación hacia el servidor. Todo este proceso se muestra en la figura 4.1.

4.2 Identificación y Autenticación

En este caso la autenticación de usuario para la aplicación móvil es el proceso de determinar si una persona está autorizada para llevar a cabo una acción dada. La identificación y autenticaciones es considerada una de las primeras líneas de defensa para la mayoría de las aplicaciones, esto con el objetivo de impedir el acceso no autorizados a la información en el sistema móvil.

El paso normal de autenticación del sistema consta de los siguientes pasos:

1. El usuario solicita al sistema.
2. El sistema le solicita al usuario que inserte su nombre de usuario y contraseña.
3. El usuario realiza su identificación.
4. El sistema valida según sus reglas los datos aportados por el usuario y verificar si son los válidos para dar acceso al sistema o no.

El paso que se propone en el sistema móvil para el acceso es el siguiente incorporando el envío de la dirección MAC del dispositivo:

1. El usuario solicita al sistema.
2. El sistema le solicita al usuario que inserte su nombre de usuario y contraseña.

3. El usuario realiza su identificación.
4. El sistema envía los datos de usuario, contraseña y dirección MAC al servidor.
5. El sistema valida según sus reglas los datos aportados por el usuario.

4.3 Encriptación con MD5

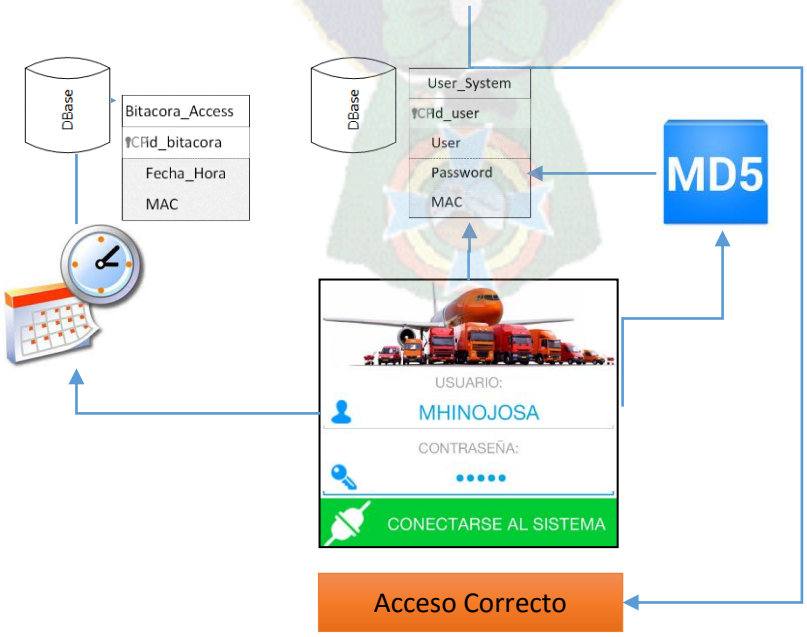
La información que se guarda en la base de datos específicamente en la tabla **User_System** en el servidor Web se encripta y se guarda en la base de datos. En el sistema se utilizará esta función para encriptar la contraseña y así tener un sistema más seguro. La codificación MD5 de 128 bits es representada típicamente como un numero de 32 dígitos hexadecimales por lo que la aplicación móvil envía los datos de usuario y password esta última para ser convertido con la función **computeMD5Hash** y poder ser comparada en la base de datos de la tabla donde están registrados los usuarios con acceso al sistema móvil**.**

4.4 Auditoria

Una de las formas de tener un control de acceso o una auditoria en el acceso y uso de la aplicación móvil es el obtener y guardar en la base de datos un registro de accesos al sistema de cada dispositivo almacenando su dirección MAC con fecha y hora. Esto ayudara a realizar una auditoría de accesos realizados por cada dispositivo en su horario y fechas que se vayan registrando en la base de datos.

En un control y seguimiento del sistema se implementa la siguiente figura:

Figura 4.1: Seguridad del Sistema



Fuente: [Elaboración Propia]

4.5 Análisis Costo-Beneficio del Sistema

La técnica de análisis de Costo Beneficio tiene como objetico fundamental proporcionar una medida de la rentabilidad del proyecto, mediante la comparación de los costos previstos con los beneficios esperados en la realización del proyecto.

4.6 Modelo Base de Estimación COCOMO

4.6.1 Punto Objeto

El procedimiento para determinar Puntos Objeto en un proyecto software se resume en:

Determinar Cantidad de Objetos: Estimar la cantidad de pantallas, reportes, componentes que contendrá la aplicación.

Clasificar cada instancia de un objeto según sus niveles de complejidad (simple, media o difícil).

Dar el peso a cada objeto según el nivel de complejidad. Los pesos reflejan el esfuerzo relativo requerido para implementar una instancia de ese nivel de complejidad.

Determinar la cantidad de Puntos Objeto, sumando todos los pesos de las instancias de los tipos de objetos especificados.

La utilidad de la presente técnica es la siguiente:

Tabla 4.1: Tabla de ponderación

Para Pantallas			
Cantidad de Vistas Contenidas	Cantidad y fuente de las tablas de datos		
	Total < 4 (< 2 servidor < 3 cliente)	Total < 8 (< 2 - 3 servidor < 3 - 5 cliente)	Total 8 + (> 3 servidor < 5 cliente)
< 3	Simple	Simple	Media
3 - 7	Simple	Media	Difícil
> 8	Media	Difícil	Difícil
Para Reportes			
Cantidad de Vistas Contenidas	Cantidad y fuente de las tablas de datos		
	Total < 4 (< 2 servidor < 3 cliente)	Total < 8 (< 2- 3 servidor < 3-5 cliente)	Total 8 + (> 3 servidor < 5 cliente)
0 o 1	Simple	Simple	Media
2 o 3	Simple	Media	Difícil
4 +	Media	Difícil	Difícil

El proyecto de acuerdo a sus características se clasifica en el siguiente modo de desarrollo:

- **Orgánico:** proyectos relativamente sencillos, menores de 50 KDLC líneas de código, en los cuales se tiene experiencia de proyectos similares y se encuentran en entornos estables.
- **Semi-acoplado / Modelo intermedio:** proyectos intermedios en complejidad y tamaño (menores de 300 KDLC), donde la experiencia en este tipo de proyectos es variable, y las restricciones intermedias. Para este caso el modelo intermedio será el que se ajusta para una estimación con bastante estimación.
- **Empotrado:** proyectos bastante complejos, en los que apenas se tiene experiencia y se engloban en un entorno de gran innovación técnica. Además, se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

Así pues, las fórmulas serán las siguientes:

E = Esfuerzo = a KLDC ^{e} * FAE (persona x mes)

T = Tiempo de duración del desarrollo = c Esfuerzo ^{d} (meses)

P= Personal = E/T (personas)

En nuestro caso las líneas por cada PF equivalen a **53 líneas lógicas** de acuerdo a los factores de diferentes lenguajes en este caso **JAVA** es el lenguaje que se utiliza en **ANDROID STUDIO**.

KLDC= (PF * Líneas de código por cada PF)/1000

KLDC= (261,36*53)/1000= **<u>13,852 KDLC</u> < 50 KLDC = PROYECTO NO MUY COMPLEJO**

PROYECTO SOFTWARE	a	e	c	d
Orgánico	**3,2**	**1,05**	**2,5**	**0,38**
Semi-acoplado	3,0	1,12	2,5	0,35
Empotrado	2,8	1,20	2,5	0,32

- **Cálculo del esfuerzo del desarrollo:**

  E = a KLDC e * FAE = 3,2 * (13,852)^1,05 * 0,53508480 = **27,04 personas /mes**

- **Cálculo tiempo de desarrollo:**

  T = c Esfuerzo d = 2,5 * (27,04)^0,38 = **8,75 meses**

- **Productividad:**

  PR = LDC/Esfuerzo = 5200/27,04= **192 ,30 LDC/personas mes**

- **Personal promedio:**

  P = E/T = 27,04/8,75 = **3,09 personas**

Según estas cifras será necesario un equipo de 3 personas trabajando alrededor de 8 meses, pero puesto que el desarrollo del proyecto debe realizarse en un plazo 4 meses, incrementaremos a 7 personas el número de personas del equipo de proyecto (ya que 27,04/4 nos da alrededor de este resultado).

Con estos resultados tendremos un equipo formado por 1 Jefe de proyecto, 2 Analistas, 3 programadores y 1 Responsable de calidad.

- **Costo de Facilidades**

  El costo de facilidades es de 0 USD porque se cuenta con un cuarto de computadoras, oficina para el personal de operación, la gente de mantenimiento y el personal usuario.

- **Estudio del Sistema**

  Se toma en cuenta que un ingeniero de software percibe un salario de 500 a 700 USD

  El costo de estimación de proyecto es:

  - Tiempo equivalente a 4 meses

  Costo del estudio del sistema por los 4 meses es 2000 USD.

- **Costos de Hardware**

  La institución deberá invertir en la compra de dispositivos móviles de mediano costo para el personal operativo.

  - Costo estimado por dispositivos = 170 USD
  - Cantidad que se requiere es = 6 unidades total 1,020 USD
  - Material de apoyo y de escritorio es de 120 USD

- **Costo del Software**

  LDC son líneas de código estimadas de código

FUNCION / MODULO	**LDC Estimado**
Rastreo de guías	800
Mis recojos	1000
Mis entregas	2000
POD	1000
Noticias TNT	400
TOTAL	**5200** LDC Estimadas

- **Costo Personal**

Cargo o Descripción	**Salarios en USD**
Ingeniero en Software	500
Programador	170
Transcriptor	70

  - El costo asignado al ingeniero y programador es = 0 USd por tratarse de un proyecto de grado.

- Costo de mano de obra = 70 USD

- **Costo Total operacional del Proyecto**

Descripción	**Monto en USD**
Costo de Hardware	1020
Costo de Software	0
Costo de Mano de obra	70
Costo de Apoyo	120
Costo de facilidades	0
TOTAL	**1210 USD**

4.7 Beneficios

En este caso los beneficio que brinda el sistema es principalmente operativo y de logística, además que la mejora en los tiempos de respuesta y de actualización de las entregas.

El siguiente cuadro refleja la actualización de los envios

TAREAS	**Sin el Sistema Móvil**	**Tiempo**	**Sistemas Móvil**	**Tiempo**
Actualización de entrega	La actualización de cada guia se realiza en el sistema de seguimiento en la página web www.tnt.com.bo	8 Hrs o al día siguiente	La actualización se realiza inmediatamente al realizar la entrega	Inmediato
Actualización de estatus de Entrega	La actualización de cada guia se realiza en el sistema de seguimiento en la página web www.tnt.com.bo	8 Hrs o al día siguiente	La actualización se realiza inmediatamente al realizar la entrega	Inmediato
Consulta de Recojo	Las consultas se realizan mediante	Cada Hora o por turnos de la tarde	La consulta se realiza en línea	Inmediato Según

	llamadas telefónicas dictadas por el Dispatcher		y de acuerdo a requerimiento	requerimiento
Rastreo de Guia	El rastreo de guia no lo realiza el mensajero, debe realizar una llamada a Servicio al cliente	No se aplica	El rastreo se realiza mediante le módulo de rastreo según requerimiento	Inmediato Según requerimiento
Prueba de entregas	Las pruebas de entregas se realizan de forma manual escaneando las mismas para enviar al cliente	Mucha demora en los tiempos de respuesta	El módulo de POD carga las fotografía de la prueba de entrega enlazada a la página web	Inmediato Según requerimiento

Según se observa la actualización y consultas se realiza inmediatamente de acuerdo a la cobertura de cada zona.

CAPITULO 5

5. Conclusiones y Recomendaciones

5.1 Conclusiones

El desarrollo de la aplicación móvil, se realizó tomando en cuenta los requerimientos y la mejor forma de optimizar la actualización de la información de las entregas y recolecciones, de acuerdo a los objetivos específicos se obtuvo los siguientes resultados:

- ✓ Se logró controlar las entregas de correspondencia aplicando el modelo de colas de prioridad.
- ✓ Se implementó y adecuo el modelo de colas con prioridad para realizar consultas para las entregas de todos los envios según al tiempo, tipo de servicio y/o tipo de cliente.
- ✓ Se implementó la aplicación móvil con la lectura de código de barras de la guia y confirmación de entrega tomando una fotografía de la guía con la firma del consignatario que serán actualizados para ser consultado por el número de guía.
- ✓ Se redujo el tiempo de respuesta en consultas con disponibilidad de información actualizada de cada envío desde internet.
- ✓ Se pudo agilizar el proceso de actualización de cada envío con el estatus que corresponde a cada situación que se presenta en cada entrega.
- ✓ Se pudo controlar el acceso al sistema Android mediante id personales para cada Courier.
- ✓ La aplicación generar consultas según criterio de propiedades de cada tipo de envío.

También aparte cumplir con los objetivos específicos se pude concluir lo siguiente:

- ✓ El sistema móvil trabaja de acuerdo a lo esperado, logrando la optimización de consumo de datos en los planes de internet móvil.
- ✓ Las metodologías utilizadas para aplicaciones móviles aún están en periodo de madurez y constante actualización donde se debe aún tomar algunas cosas prácticas de metodologías orientadas a la Web.

5.2 Recomendaciones

Las recomendaciones que se logran dar a partir del manejo del sistema son:

- ✓ Se recomienda la integración del sistema con otros sistemas que maneja la empresa en el área operativa y de gestión comercial para obtener datos de movimiento y control de envios de los clientes.

- ✓ Complementar en el desarrollo de otros módulos de control y seguimiento a la aplicación como ser de cotización de envios, tiempos de tránsito para las consultas que los clientes realizan a los mensajeros.

- ✓ Crear una aplicación de distribución en la Play Store de Google para que los clientes puedan descargarla e instalar en sus dispositivos y que realicen seguimiento desde la aplicación.

- ✓ Se recomienda realizar una concientización a los mensajeros que los planes de internet móvil deben ser utilizados únicamente para el cumplimiento de manejo y actualización de la aplicación móvil, y no así usarlo en otros programas que gastan los datos de internet como ser paginas sociales y mensajería.

- ✓ Se recomienda continuar con el proyecto y mejorar la aplicación con módulos que permitan trabajar con las áreas de ventas, atención al cliente.

Bibliografía

- **[Alan wills, 1998]**, objetos, componentes y estructuras con uml, the catalysisaproach, por desmond f. D'souza yalan c. Wills, addison wesley longman, 1998.
- **[Carvajal, J. 2008]** Metodologías Ágiles. ES. Consultado, 06 de mayo de 2015. Formato PDF. https://upcommons.upc.edu/pfc/bitstream/2099.1/5608/1/50015.pdf
- **[Cocomo ii, 2000],** b. Boehm, c. Abts, a. Brown, s. Chulani, b. Clark, e. Horowitz, r. Madachy, d. Reifer, b. Steece: software cost estimation with cocomo ii. Prentice-hall, 2000.
- **[Extreme programming explained: embrace change, 1999]**, kent beck y cynthia andres, foreword by erich gamma, segunda edicion.
- **[extreme programming, 2003],** josé carlos cortizo pérez, diego expósito gil y miguel ruiz leyv; **http://www.esp.uem.es/jccortizo/xp.pdf**, ainet solutionstechnicalreport no. 01.
- **[Introducción a los negocios en un mundo cambiante, 2004]**, de ferrel o.c., hirt geofrey, ramos leticia, adriaenséns marianela y flores miguel angel, mc graw hill, 2004, cuarta edición, pág. 282.
- **[Mobile-D, 2012],** Electronics - AGILE - Agile Software Technologies», 21-nov-2012. [En línea]. Disponible en: http://virtual.vtt.fi/virtual/agile/mobiled.html
- **[Raiz emprendedor, 2010],** artículo subido el día 22 de enero del 2010, http://www.raizemprendedor.com/definicion-de-courier-paqueteria-y-logistica/81
- **[Rossi, 1996],** rossi, g. An object-oriented method for designing hypermedia applications. Brasil, puc-rio, 1996
- **[Schwabe d, 1995],** schwabe d., r. G.the object-oriented hipermedia design model. Communications of the acm, 1995.
- **[TNT news, 2015],** revista TNT news, edición europea - enero 2015 - numero 06.
- **[TNT web, 2016],** web site mundial de TNT – 2016 – enlace a historia y orígenes de la empresa http://www.TNT.com/corporate/en/site/home/about_us/history.html

- **[Web1, 2009],** university of southern california center for software engineering http://sunset.usc.edu/csse/research/cocomoii/cocomo_main.html. Septiembrede 2009.