

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



TESIS DE GRADO

**“APLICACIÓN DE LOCALIZACIÓN Y CONTROL
DE DISPOSITIVOS SMARTPHONE ANDROID”**

**PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMÁTICA
MENCIÓN: INGENIERÍA DE SISTEMAS INFORMÁTICOS**

POSTULANTE : CONDORI CALIZAYA IVAN ARSENIO
TUTOR METODOLÓGICO : LIC. FREDDY MIGUEL TOLEDO PAZ
ASESOR : M.SC. CARLOS MULLISACA CHOQUE

LA PAZ – BOLIVIA

2014



**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

Con todo mi cariño y mi amor para la persona que hizo todo en la vida para que yo pudiera lograr mis sueños, por motivarme y darme la mano cuando sentía que el camino se terminaba, a ti por siempre mi corazón y mi agradecimiento.

Mi querida mami, Porfiria

A mí querida familia

A mi papa Arsenio y mis hermanos José Antonio y Abraham, quienes con cariño y amor me apoyaron en todo momento, brindándome su comprensión y confianza, dándome las fuerzas suficientes para alcanzar esta meta.

Y mis dos amores mi esposa Mónica y mi hija Melany que son la razón de mi vida por estar siempre ahí cuando más las necesito.

Los quiero Mucho.....

AGRADECIMIENTOS

Un Trabajo de Tesis de Grado, constituye un gran esfuerzo personal para alcanzar los objetivos propuestos, dicho trabajo no sería posible de realizar sin la colaboración, apoyo, orientación, consejos, sugerencias, conocimientos y experiencias de mis docentes y amigos.

Agradezco sinceramente al Lic. Freddy Miguel Toledo Paz quién me dedico su tiempo, paciencia y comprensión al realizar el seguimiento de esta Tesis, orientándome a través de su conocimiento y sus sugerencias para la culminación del trabajo.

Al M.Sc. Carlos Mullisaca Choque, por su apoyo, comprensión, paciencia y dedicación incondicional, por sus sugerencias, observaciones, seguimiento constante, conocimientos y experiencias brindadas en el transcurso del presente trabajo y por todo el tiempo dedicado en cada una de sus revisiones.

También agradezco a todo el plantel Docente que durante todo este tiempo instruyó y capacito, transmitiéndome sus conocimientos y experiencias, de igual forma al plantel Administrativo, Jefatura de Carrera, Kárdex, Laboratorio (Lasin) de Informática y Biblioteca por la documentación y apoyo brindado,

RESUMEN

La introducción y crecimiento en el mercado de los Smartphone en la población es muy alta, estos dispositivos nos proporcionan nuevas funcionalidades, tratándose a tal punto de considerarlos pequeños ordenadores. Android se ha impuesto como el principal sistema operativo, en cuanto a cuota de mercado, de la presente generación de teléfonos móviles. En este trabajo se ha desarrollado una aplicación permite controlar y localizar estos dispositivos, dándonos a conociendo su ubicación en determinados intervalos de tiempo, lo cual puede ser utilizado para emitir alarmas si un usuario sale de un determinado perímetro de control.

A lo largo de este documento se va a dar a conocer una breve historia de la telefonía móvil y se va a describir el proceso de diseño y desarrollo de la aplicación mencionada.

El desarrollo comprende, no sólo una aplicación móvil que se instalará y ejecutará en el teléfono móvil, sino también una página web con interface Google Maps que interprete la información recibida desde el teléfono.

ABSTRACT

The introduction and growth in the smartphone market in the population is very high, these devices provide us with new features, the case to the point of considering small computers. Android has emerged as the main operating system, in terms of market share, the current generation of mobile phones. In this work we have developed an application allows you to control and locate these devices, giving us to know their location in specific time intervals, which can be used to issue alarms when a user exits a given perimeter control.

Throughout this document is to give out a brief history of the mobile phone and will describe the process of designing and developing the application mentioned.

The development includes not only a mobile application that will install and run on the mobile phone, but also a site that interprets the information received from the phone

ÍNDICE DE CONTENIDO

CAPITULO I

1. INTRODUCCIÓN.....	14
2. ANTECEDENTES.....	15
2.1. EVOLUCIÓN DE LA TECNOLOGÍA MÓVIL.....	15
2.2. PROYECTOS SIMILARES.....	18
2.2.1. RASTREO POR GPS.....	18
2.2.2. GPS SHARE.....	18
3. PLANTEAMIENTO DEL PROBLEMA.....	19
3.1. PROBLEMA GENERAL.....	19
3.2. PROBLEMAS ESPECÍFICOS.....	19
4. OBJETIVOS.....	20
4.1. OBJETIVO GENERAL.....	20
4.2. OBJETIVO ESPECIFICO.....	20
5. HIPÓTESIS.....	20
6. JUSTIFICACIÓN.....	21
6.1. JUSTIFICACIÓN ECONÓMICA.....	21
6.2. JUSTIFICACIÓN SOCIAL.....	21
6.3. JUSTIFICACIÓN TÉCNICA.....	21
7. ALCANCES.....	23
8. LIMITES.....	23
9. APORTES.....	24

CAPITULO II

10. MARCO TEÓRICO.....	26
------------------------	----

10.1. METODOLOGÍA DE DESARROLLO ÁGIL.....	26
10.2. PROCESO UNIFICADO ÁGIL (AUP).....	28
10.2.1. DISCIPLINA DE LA METODOLOGÍA PROCESO UNIFICADO ÁGIL (AUP).....	29
10.2.2. PRINCIPIOS AUP.....	31
11. METODOLOGÍA DE DESARROLLO UWE.....	32
11.1. ACTIVIDADES DE MODELADO DE UWE.....	33
11.2. MODELO LÓGICO CONCEPTUAL.....	34
11.3. MODELO DE NAVEGACIÓN.....	34
11.4. MODELO DE PRESENTACIÓN.....	35
11.4.1. INTERACCIÓN TEMPORAL.....	35
11.4.2. ESCENARIO WEB.....	35
11.4.3. DIAGRAMAS.....	35
11.5. FASES DE LA UWE.....	36
11.5.1. CAPTURA ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS.....	36
11.5.2. DISEÑO DEL SISTEMA.....	36
11.5.3. CODIFICACIÓN DEL SOFTWARE.....	36
11.5.4. PRUEBAS.....	37
11.5.5. LA INSTALACIÓN O FASE DE IMPLEMENTACIÓN.....	37
11.5.6. MANTENIMIENTO.....	37
12. HERRAMIENTAS A UTILIZAR.....	37
12.1. ELECCIÓN DE LA PLATAFORMA DE TRABAJO.....	37
12.1.1. PLATAFORMAS MÓVILES.....	37
12.2. HERRAMIENTAS DE DESARROLLO A USAR.....	40
12.2.1. ECLIPSE.....	40
12.2.2. ANDROID DEVELOPMENT TOOLS (ADT).....	42
12.3. GPS.....	42
12.3.1. CARACTERÍSTICAS Y USOS.....	42

12.4.	PLATAFORMA DE TRABAJO.....	44
12.4.1.	ANDROID.....	44
12.4.2.	OPEN HANDSET ALLIANCE.....	45
12.4.3.	HISTORIA DE ACTUALIZACIONES.....	45
12.4.4.	VERSIONES.....	46
12.4.5.	ARQUITECTURA DE ANDROID.....	47
12.4.5.1.	KERNEL DE LINUX.....	48
12.4.5.2.	BIBLIOTECAS.....	48
12.4.5.3.	ENTORNO DE EJECUCIÓN.....	48
12.4.5.4.	MARCO DE APLICACIÓN.....	49
12.4.5.5.	APLICACIÓN.....	49
12.4.5.6.	ESTRUCTURA DE UNA APLICACIÓN.....	49
12.4.5.7.	ACTIVITIES.....	49
12.4.5.8.	INTENTS.....	50
12.4.5.9.	VIEWS.....	50
12.4.5.10.	SERVICES.....	50
12.4.5.11.	CONTENT PROVIDERS.....	50
12.4.5.12.	MANIFEST.....	52
12.4.5.13.	BROADCAST RECEIVERS.....	52
12.5.	GOOGLE MAPS.....	52
12.6.	OTRAS HERRAMIENTAS.....	52

CAPITULO III

13.1.	INTRODUCCIÓN.....	55
13.2.	FASE DE COMIENZO.....	55
13.2.1.	DISCIPLINA DEL MODELADO.....	56
13.2.1.1.	ANÁLISIS DE REQUISITOS.....	56
a.	DEFINIR ACTORES.....	56

b.	DEFINIR RELACIONES ENTRE ACTORES.....	57
c.	DEFINIR RELACIONES ENTRE ACTORES.....	57
13.2.2.	DISCIPLINA DE IMPLEMENTACIÓN.....	58
13.2.2.1.	CASO DE USO POR ACTOR (ACTOR ADMINISTRADOR).....	58
13.2.2.2.	CASO DE USO POR ACTOR (ACTOR TERMINAL MÓVIL).....	63
13.3.	FASE DE ELABORACIÓN.....	66
13.3.1.	DISCIPLINA DE MODELADO.....	66
13.3.1.1.	CAPA CONCEPTUAL.....	66
a)	DISEÑO CONCEPTUAL.....	66
b)	DIAGRAMA ENTIDAD RELACIÓN.....	67
c)	MODELO FÍSICO.....	68
d)	DISEÑO DE NAVEGACIÓN.....	69
e)	DIAGRAMA GENERAL DEL FUNCIONAMIENTO DEL SISTEMA COMPLETO.....	70
f)	DISEÑO DE LA INTERFACE.....	71

CAPITULO IV

14. CAPITULO IV EXPERIMENTACIÓN Y

	ANÁLISIS DE RESULTADOS.....	77
14.1.	EXPERIMENTACIÓN Y ANÁLISIS DE RESULTADOS.....	77
14.1.1.	DETERMINACIÓN DE LA MUESTRA.....	77
14.1.2.	PROCESO DE PRUEBA CON LA APLICACIÓN MÓVIL...	79
14.1.3.	EVALUACIÓN DE RESULTADOS.....	79
14.1.4.	PRUEBA DE HIPÓTESIS.....	82

CAPITULO V

15. CONCLUSIONES.....	86
16. RECOMENDACIONES.....	87
BIBLIOGRAFÍA.....	88
ANEXOS.....	90

ÍNDICE DE FIGURAS

FIGURA 1.1: HANDIE TALKIE H12-16 FABRICADO POR MOTOROLA.....	15
FIGURA 2.1: DYNATAC 8000X.....	15
FIGURA 2.2: LA EVOLUCIÓN DE LA TECNOLOGÍA DEL MÓVIL Y SUS CUATRO GENERACIONES.....	16
FIGURA 2.3 ESQUEMA GENERAL DE UNA METODOLOGÍA ÁGIL....	27
FIGURA 2.4 DISCIPLINA DE LA METODOLOGÍA PROCESO UNIFICADO ÁGIL.....	29
FIGURA 2.5: PORCENTAJE DE TELÉFONOS INTELIGENTES VENDIDOS SEGÚN SU SISTEMA OPERATIVOS.....	39
FIGURA 2.6: EL HTC DREAM PRIMER TELÉFONO EN INCORPORAR ANDROID.....	44
FIGURA 2.7: EVOLUCIÓN DE LAS VERSIONES DE ANDROID.....	46
FIGURA 2.8: ARQUITECTURA DE ANDROID.....	47
FIGURA 2.9: ESTRUCTURA DE UNA APLICACIÓN MÓVIL ANDROID.....	51
FIGURA 3.1: ANÁLISIS DE CASO DE USO.....	57
FIGURA 3.2: ANÁLISIS DE CASO DE USO ADMINISTRADOR.....	58
FIGURA 3.3: DISEÑO CONCEPTUAL.....	66

FIGURA 3.4: DIAGRAMA ENTIDAD RELACIÓN.....	67
FIGURA 3.5: DISEÑO MODELO FÍSICO.....	68
FIGURA 3.6: DISEÑO DE NAVEGACIÓN.....	69
FIGURA 3.7: DIAGRAMA GENERAL DEL FUNCIONAMIENTO DEL SISTEMA COMPLETO.....	70
FIGURA 3.8: INGRESANDO COMO ADMINISTRADOR.....	71
FIGURA 3.9: PANTALLA DE INICIO.....	72
FIGURA 4.0: PANTALLA DE INICIO.....	72
FIGURA 4.1: PANTALLA GENERA REPORTE.....	73
FIGURA 4.2: RASTREO DE TERMINALES MÓVILES.....	73
FIGURA 4.3: RASTREO DE TERMINALES MÓVILES MAS RESTRICCIÓN DE ZONA.....	74
FIGURA 4.4: REGISTRO DE TERMINALES MÓVILES.....	75
FIGURA 4.5: RESPUESTA A FALLA DE REGISTRO DE TERMINALES MÓVILES.....	75

ÍNDICE DE TABLAS

TABLA 1.1 JUSTIFICACIÓN TÉCNICA.....	22
TABLA: 2.1: TABLA COMPARATIVA DE PLATAFORMAS MÓVILES.....	38
TABLA 3.1: DEFINICIÓN DE ACTORES.....	56
TABLA 3.2 INSTALAR APLICACIÓN EN TERMINAL MÓVIL.....	59
TABLA 3.3 CONFIGURACIÓN DE LA APLICACIÓN MÓVIL.....	60
TABLA 3.4 CONFIGURACIÓN DE LA APLICACIÓN MÓVIL.....	61

TABLA 3.5 CONFIGURACIÓN DE LA APLICACIÓN MÓVIL.....	62
TABLA 3.6 CONFIGURACIÓN DE LA APLICACIÓN MÓVIL.....	64
TABLA 3.7 CONFIGURACIÓN DE LA APLICACIÓN MÓVIL.....	65
TABLA 4.1: ESCALA DE EVALUACIÓN DE LA APLICACIÓN MÓVIL.....	79
TABLA 4.2: RESULTADOS DE LA EVALUACIÓN DE LA APLICACIÓN MÓVIL.....	80
TABLA 4.2: COMPARACIÓN DE RESULTADOS DE LA EVALUACIÓN DE LA APLICACIÓN MÓVIL.....	80
TABLA 4.3: MEDIAS DE TENDENCIA CENTRAL ACERCA DE LA INTERACTIVIDAD.....	83
TABLA 4.4: RESULTADOS DEL CÁLCULO DE VALOR T.....	83
TABLA 4.5: DISTRIBUCIÓN T DE STUDENT PARA 18 GRADOS DE LIBERTAD.....	85

ÍNDICE DE ACRÓNIMOS

- API** Application Programming Interface (Interfaz de programación de aplicaciones)
- DDMS** Dalvik Debug Monitor Server (Monitor de depuración de la maquina Dalvik)
- EDGE** Enhanced Data rates GSM of Evolution (Evolución de GSM)
- GPS** Global Positioning System (Sistema de posicionamiento global)
- GSM** Groupe Spécial Mobile (Sistema global para las comunicaciones móviles)
- HSPA** High Speed Packet Access (Acceso por paquetes a alta velocidad)
- HTML** HyperText Markup Language (Lenguaje de marcado de hipertexto)
- IP** Internet Protocol (Protocolo de Internet)
- LTE** Long Term Evolution (Estándar de cuarta generación móvil)
- NFC** Near Field Communication (Comunicación por campo cercano)
- OHA** Open Handset Alliance (Alianza comercial formada por Google entre otros)
- PDA** Personal Digital Assistant (Asistente personal digital)
- SDK** Software Development Kit (Kit de desarrollo software)
- SMS** Short Message Service (Servicio de mensajes cortos)
- TCP** Transmission Control Protocol (Protocolo de control de transmisión)
- UDP** User Datagram Protocol (Protocolo basado en intercambio de datagramas)
- UMTS** Universal Mobile Telecommunications System (Estándar de tercera generación)
- Wi-Fi** Wireless Fidelity (Mecanismo de conexión inalámbrica)



CAPITULO I

Perfil

1. INTRODUCCIÓN

A lo largo de la historia, las sociedades han evolucionado a través del contacto entre sus miembros. El comercio y la comunicación han sido indispensables en ese proceso.

En la actualidad para muchas personas y empresas dedicadas al transporte, la mensajería y otros relacionados con la movilidad de personal, contar con dispositivos que permiten conocer la ubicación exacta de sus familiares, amigos, empleados es una gran ayuda para su organización logística y optimización de recursos.

La historia del teléfono móvil se remonta a los inicios de la Segunda Guerra Mundial, donde ya se veía que era necesaria la comunicación a distancia de un lugar a otro, es por eso que la compañía Motorola creó un equipo llamado Handie Talkie H12-16 (véase FIGURA 1.1), que es un equipo que permite el contacto con las tropas vía ondas de radio cuya banda de frecuencias en ese tiempo no superaban los 600 kHz.

En la actualidad los teléfonos móviles de última generación, llamados Smartphone o teléfonos inteligentes. No se limitan a comunicar a sus usuarios por voz o por mensajes de texto. Incluyen servicios y programas que los acercan más a los ordenadores portátiles que a los móviles. Cumplen todas las funciones de una agenda u organizador personal y pueden incluir hasta sistemas de GPS¹.

Es así que haciendo uso de esta tecnología móvil específicamente la plataforma Android se presenta esta tesis de grado en el cual se desarrollara una aplicación de localización y control de dispositivos Smartphone Android

¹ GPS (Sistema de Posicionamiento Global) constituido por 24 satélites y utiliza la triangulación para determinar en todo el globo la posición con una precisión de más o menos metros.



FIGURA 1.1: Handie Talkie H12-16 fabricado por Motorola
FUENTE: (Wikipedia junio 2014)

2. ANTECEDENTES

2.1. EVOLUCIÓN DE LA TECNOLOGÍA MÓVIL



FIGURA 2.1: Dynatac 8000x
FUENTE: (Wikipedia junio 2014)

El teléfono móvil Dynatac 8000x fue desarrollado en el año 1983 por Motorola, siendo el primer teléfono móvil del mundo y fue puesto a la venta el 13 de marzo del mismo año. Podemos mencionar que pesaba 800 gramos y medía 33 por 4,5 por 8,9 centímetros. En la fecha de salida al mercado del producto, el terminal costaba 3.995 dólares estadounidenses y su batería tenía únicamente la autonomía de una hora en conversación. Un año más tarde (1984) 300.000 usuarios habían comprado el terminal. Este primer paso de la

revolución móvil vino de la mano de Rudy Krolopp².

² Rudy Krolopp ingeniero de Motorola, diseñó el modelo del teléfono móvil Dynatac 8000x

Por esta época, estas terminales estaban considerados como un lujo y estaban orientados principalmente al sector empresarial (véase FIGURA 2.1).



FIGURA 2.2: La evolución de la tecnología del móvil y sus cuatro generaciones
FUENTE: (<http://www.eurekamovil.es/>)

No fue hasta la llegada del estándar GSM (Groupe Spécial Mobile) cuando el teléfono móvil se popularizó entre la población. Esto fue debido a que estas terminales los terminales podían funcionar en cualquier región del mundo, provocando con ello una reducción del coste de los terminales y permitiendo el acceso a los consumidores.

Fue también de gran acierto la implementación de los SMS (el servicio de mensajes cortos de texto) y la definición de un número de emergencias único a nivel global. Con las bases ya sentadas gracias a GSM, la evolución de los estándares fueron mejorando las características de la red, aumentando las

velocidades de comunicación, la cobertura, etc. De este modo fueron apareciendo estándares como: GPRS, EDGE, UMTS o HSPA, estas últimas pertenecientes a la denominada como tercera generación de la telefonía móvil.

Estos avances en las telecomunicaciones han sido apareciendo progresivamente. Se empezó con los primeros accesos a Internet con la tecnología WAP, y continuó con la aparición del correo electrónico en los terminales móviles impulsado por BlackBerry. Posteriormente la llegada de los Smartphone proporcionó una experiencia en Internet que no era capaz de ofrecernos las tecnologías anteriores (véase FIGURA 2.2).

Actualmente, estamos en la cuarta generación de telefonía móvil, cuya abanderada es LTE Advanced (Long Term Evolution Advanced).

Complementariamente a este avance ha sido necesario el desarrollo de nuevas baterías que nos permitan una mayor duración al menor tamaño posible, además la aparición de las nuevas pantallas de gran tamaño y grandes colores nos permite explotar todas las capacidades multimedia de los dispositivos. Todo esto ha convertido a los teléfonos móviles en un elemento imprescindible en la vida cotidiana.

El avance de la tecnología ha conseguido que ahora se incorporen funcionalidades tales como el correo electrónico, la reproducción de MP3, los juegos, la fotografía y el video digital, video llamadas, la localización GPS, la navegación por Internet y actualmente se está trabajando en el uso de la tecnología NFC³.

³ **NEAR FIELD COMMUNICATION** (NFC), en castellano: (Comunicación de Campo Cercano) es una tecnología de comunicación inalámbrica, de corto alcance y alta frecuencia que permite el intercambio de datos entre dispositivos

2.2. PROYECTOS SIMILARES

Estas aplicaciones pueden ser encontradas en Google Play

2.2.1. RASTREO POR GPS

GPS que sigue el pro utiliza la tecnología GPS Tracker estado-of-the-art para localizar con rapidez y precisión las personas que te importan, proporcionando actualizaciones de localización en tiempo real utilizando sistemas de navegación GPS. La aplicación cuenta con un mapa GPS con los iconos de la familia y amigos en sus ubicaciones exactas. Los padres aman el seguimiento pro GPS, ya que proporcionan información actualizada al minuto sobre su hijo (s) 's paradero. Es una herramienta útil para monitorear la ubicación de los miembros mayores de la familia, ayudando a mantenerlos a salvo. Además de ser un buscador de gran pueblo, la localización por GPS pro también rastrea los teléfonos que incluyen dispositivos móviles, teléfonos inteligentes, dispositivos más antiguos y los androides.

2.2.2. GPS SHARE

¿Alguna vez tratando de reunirse con amigos en un gran espacio público, pero no podido encontrar uno al otro? He creado GPS Compartir para apenas tal escenario. GPS Compartir le permite enviar de forma privada un enlace de Google Maps de su ubicación para quien quieras, cuando quieras, como quieras. Enviar un mensaje de texto para encontrarse con amigos en un estadio o enviarse por correo electrónico un recordatorio de dónde estacionó en el aeropuerto. Publicar actualizaciones de viaje por carretera a Twitter.

A diferencia de Latitud y muchos otros servicios, GPS Compartir no se ejecuta en segundo plano y no comparte su ubicación con un tercero. Se usa cuando usted lo desee.

GPS Compartir es libre y software de código abierto y no tiene publicidad. Conexión a Internet sólo se utiliza para descargar los azulejos de Google Maps.

3. PLANTEAMIENTO DEL PROBLEMA

3.1. PROBLEMA GENERAL

¿Mediante una Aplicación Android se podrá realizar la localización y control de dispositivos Smartphone Android?

3.2. PROBLEMAS ESPECÍFICOS

- Si bien existen muchas aplicaciones para ubicar Smartphone Android estas no registran ni almacenan esta información.
- No existen aplicaciones Smartphone Android que emitan alertas para cuando estos salen de una determinada área
- No existen aplicaciones que limiten a una determinada área el desplazamiento de los Smartphone Android
- No existe una aplicación específicamente diseñada para la ciudad de La Paz

4. OBJETIVOS

4.1. OBJETIVO GENERAL

Implementar una aplicación móvil y una página web con interface GoogleMap para la localización y control de dispositivos Smartphone Android.

4.2. OBJETIVO ESPECIFICO

- Implementar una página Web con interface Google Maps para visualizar en tiempo real los dispositivos Smartphone Android.
- Implementar Módulos para la localización de Smartphone Android
- Diseñar una base de datos que almacene la información de la localización y control de celulares Smartphone Android.
- Diseñar un módulo que genere automática mente alertas en el caso que una terminal Smartphone Android se salga de una determinada área de asignación de desplazamiento.
- Segmentar la ciudad de La Paz, en áreas para monitorear el desplazamiento de los dispositivos Smartphone Android.

5. HIPÓTESIS

“La implementación de una aplicación Android con interface Google Maps optimiza la localización y control de dispositivos Smartphone Android”.

6. JUSTIFICACIÓN

6.1. JUSTIFICACIÓN ECONÓMICA

Se justifica económicamente porque se cuenta con todos los recursos económicos, que en este caso no son elevados al igual que el hardware y software,

Debemos mencionar que se usara software libre que no demandara gastos por licencias como ser Android Development Tools (ADT), que es un plugin para IDE Eclipse que nos ayudara a desarrollar la aplicación móvil.

6.2. JUSTIFICACIÓN SOCIAL

Las personas que hagan uso de la aplicación podrán hacer el monitoreo y control de no solamente una sino varias terminales Smart Phones Android dando tranquilidad a las mismas y certeza que todo se está desarrollando con normalidad

6.3. JUSTIFICACIÓN TÉCNICA

Se debe mencionar que se cuentan con internet banda ancha ADSL, hosting y los medios necesarios de hardware, software así como las metodologías necesarias para realizar el presente proyecto.

Las características de los equipos móviles y de computación son detalladas en la tabla 1.1

Cantidad	Tecnología	Características
1	Core i3 de cuatro núcleo	Procesador Q6600 2.4 GHz, 2.39 GHz Memoria RAM 4 Gigas Disco Duro de 1 Tera. Tarjeta de video XFX 880 Tarjeta de red integrada
1	Smartphone Samsung Galaxy Ace	Procesador 800MHz Android OS v2.2 Froyo WI-FI, GPS MicroSD, cámara 5 megapixels Pantalla de 3.5"
1	Samsung Galaxy Tab3	Android OS, v4.1 Jelly Bean WI-FI, GPS MicroSD, cámara 5 megapixels Pantalla de 7" 600x1024
1	Router Inalámbrico Modelo TL-WR741ND TP-LINK	IEEE 802.11n de IEEE 802.11g 1 Puerto de 10/100 RJ-45 WAN 4 PUERTOS 10/100 Rango de Frecuencia 2.4 a 2.4835 GHz

Tabla 1.1 Justificación Técnica.
Fuente: [Elaboración Propia]

7. ALCANCES

El presente proyecto está diseñado para el monitoreo y control de Smart Phone Android en determinadas áreas de desplazamiento.

Se pretende implementar lo siguiente:

- Una página Web
- Una aplicación Android para terminales Smartphone
- El registro, modificación, y eliminación de terminales Smartphone a ser monitoreadas y controladas
- Un módulo de generación de alertas automáticas en el caso que una terminal Smartphone Android se salga de una determinada área de asignación de desplazamiento.
- Un Módulo de Generación de Reportes

8. LIMITES

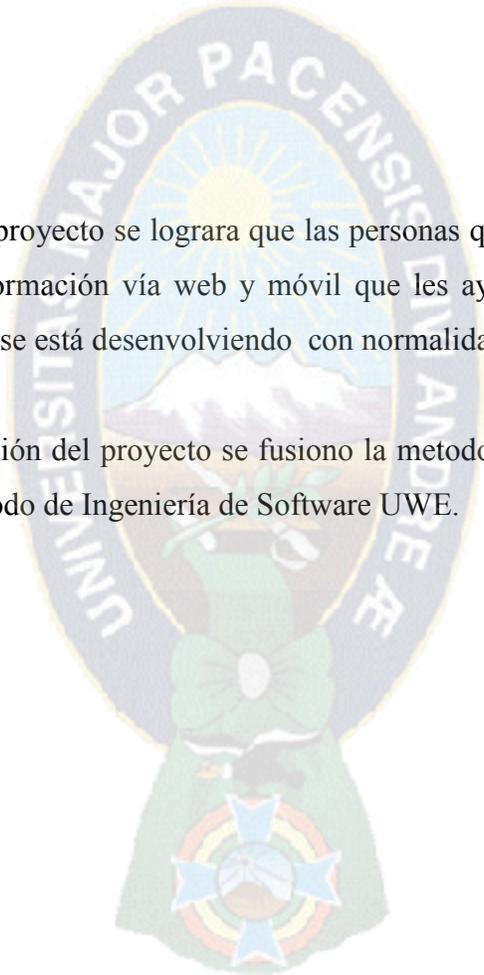
- La Aplicación está desarrollada específicamente para dispositivos Smartphone Android (WiFi, GPS) y no para otras plataformas como Apple iOS, BlackBerry, Windows Mobile y otros.
- El funcionamiento de la aplicación está limitada a las áreas urbanas que cuenten acceso a internet para tal caso la ciudad de La Paz.

- La Aplicación está orientada a realizar el control y seguimiento de terminales móviles Android indiferentemente a la persona que use esta terminal.
- La Aplicación es compatible para versiones superiores o iguales a la 2.3.3 de Android

9. APORTES

Con el presente proyecto se lograra que las personas que usen la aplicación Android cuenten con información vía web y móvil que les ayude a tener la tranquilidad y certeza que todo se está desarrollando con normalidad

Para la elaboración del proyecto se fusiono la metodología Proceso Unificado Ágil (AUP) y un método de Ingeniería de Software UWE.





CAPITULO II

Marco Teórico

10. MARCO TEÓRICO

10.1. METODOLOGÍA DE DESARROLLO ÁGIL

El desarrollo ágil de software son métodos de ingeniería del software basado en el desarrollo iterativo e incremental, donde los requisitos y soluciones evolucionan mediante la colaboración de grupos autos organizados y multidisciplinarios. Existen muchos métodos de desarrollo ágil; la mayoría minimiza riesgos desarrollando software en lapsos cortos. El software desarrollado en una unidad de tiempo es llamado una iteración, la cual debe durar de una a cuatro semanas. Cada iteración del ciclo de vida incluye: planificación, análisis de requisitos, diseño, codificación, revisión y documentación. Una iteración no debe agregar demasiada funcionalidad para justificar el lanzamiento del producto al mercado, sino que la meta es tener una «demo» (sin errores) al final de cada iteración. Al final de cada iteración el equipo vuelve a evaluar las prioridades del proyecto.

Los métodos ágiles enfatizan las comunicaciones cara a cara en vez de la documentación. La mayoría de los equipos ágiles están localizados en una simple oficina abierta, a veces llamadas "plataformas de lanzamiento" (bullpen en inglés). La oficina debe incluir revisores, escritores de documentación y ayuda, diseñadores de iteración y directores de proyecto. Los métodos ágiles también enfatizan que el software funcional es la primera medida del progreso. Combinado con la preferencia por las comunicaciones cara a cara, generalmente los métodos ágiles son criticados y tratados como "indisciplinados" por la falta de documentación técnica.

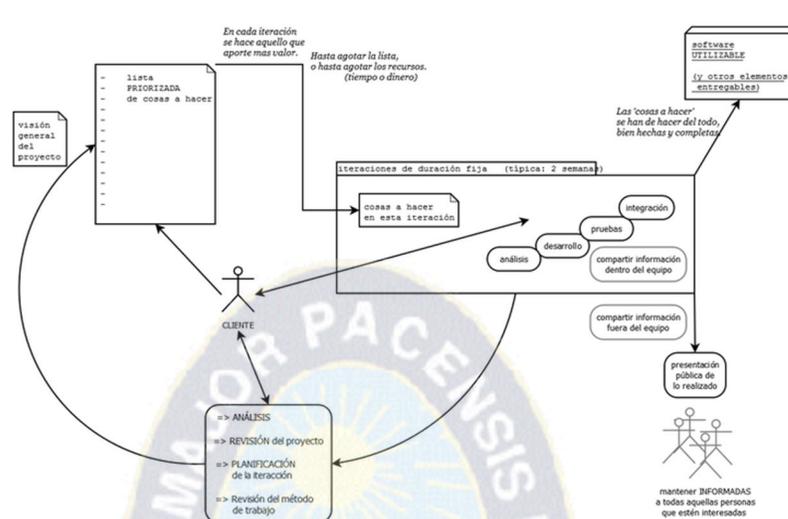


Figura 2.3 Esquema General de una metodología ágil
Fuente: [Elaboración Propia]

Metodologías ágiles:

- ✓ Extreme Programming (XP)
- ✓ Scrum
- ✓ Agile Modeling Adaptive Software Development (ASD)
- ✓ Crystal Clear
- ✓ Dynamic Systems Development Method (DSDM)
- ✓ Feature Driven Development (FDD)
- ✓ Lean Software Development (LSD)
- ✓ **Agile Unified Process (AUP)**
- ✓ Software Development Rhythms
- ✓ Agile Documentation
- ✓ ICONIX Process
- ✓ Microsoft Solutions Framework (MSF)
- ✓ Agile Data Method
- ✓ Database Refactorig

10.2. PROCESO UNIFICADO ÁGIL (AUP)

El Proceso Unificado Ágil de Scott Ambler o Agile Unified Process (AUP) en inglés es una versión simplificada del Proceso Unificado de Rational (RUP). Este describe de una manera simple y fácil de entender la forma de desarrollar aplicaciones de software de negocio usando técnicas ágiles y conceptos que aún se mantienen válidos en RUP. El AUP aplica técnicas ágiles incluyendo Desarrollo Dirigido por Pruebas (test driven development - TDD), Modelado Ágil, Gestión de Cambios Ágil, y Refactorización de Base de Datos para mejorar la productividad.

El proceso unificado (Unified Process o UP) es un marco de desarrollo software iterativo e incremental. A menudo es considerado como un proceso altamente ceremonioso porque especifica muchas actividades y artefactos involucrados en el desarrollo de un proyecto software. Dado que es un marco de procesos, puede ser adaptado y la más conocida es RUP (Rational Unified Process) de IBM.

AUP se preocupa especialmente de la gestión de riesgos. Propone que aquellos elementos con alto riesgo obtengan prioridad en el proceso de desarrollo y sean abordados en etapas tempranas del mismo. Para ello, se crean y mantienen listas identificando los riesgos desde etapas iniciales del proyecto. Especialmente relevante en este sentido es el desarrollo de prototipos ejecutables durante la base de elaboración del producto, donde se demuestre la validez de la arquitectura para los requisitos clave del producto y que determinan los riesgos técnicos.

El proceso AUP establece un Modelo más simple que el que aparece en RUP por lo que reúne en una única disciplina las disciplinas de Modelado de Negocio, Requisitos y Análisis y Diseño. El resto de disciplinas (Implementación, Pruebas, Despliegue, Gestión de Configuración, Gestión y Entorno) coinciden con las restantes de RUP.

10.2.1. DISCIPLINA DE LA METODOLOGÍA PROCESO UNIFICADO ÁGIL (AUP)

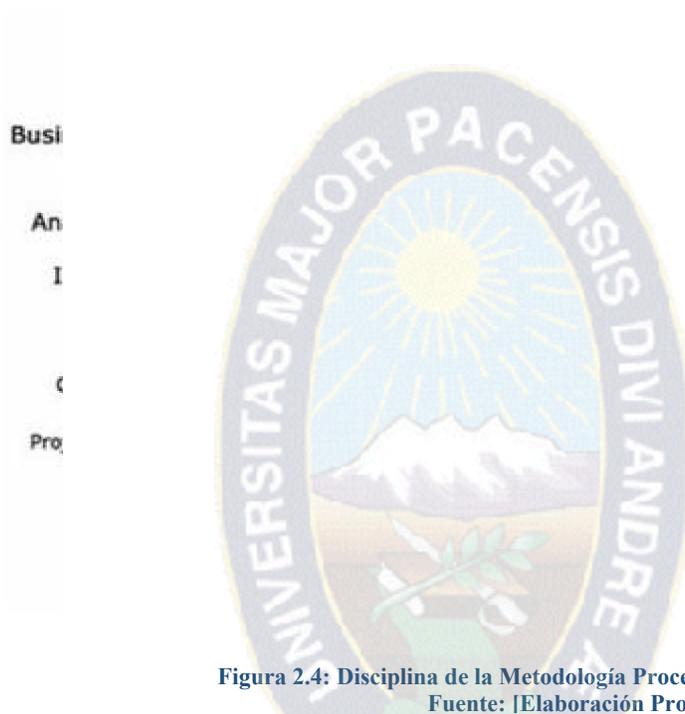


Figura 2.4: Disciplina de la Metodología Proceso Unificado Ágil (AUP)
Fuente: [Elaboración Propia]

Al

igual que en RUP, en AUP se establecen cuatro fases que transcurren de manera consecutiva y que acaban con hitos claros alcanzados:

Inception (Concepción): El objetivo de esta fase es obtener una comprensión común cliente equipo de desarrollo del alcance del nuevo sistema y definir una o varias arquitecturas candidatas para el mismo.

Elaboración: El objetivo es que el equipo de desarrollo profundice en la comprensión de los requisitos del sistema y en validar la arquitectura.

Construcción: Durante la fase de construcción el sistema es desarrollado y probado al completo en el ambiente de desarrollo.

Transición: el sistema se lleva a los entornos de preproducción donde se somete a pruebas de validación y aceptación y finalmente se despliega en los sistemas de producción.

Las disciplinas se llevan a cabo de manera sistemática, a la definición de las actividades que realizan los miembros del equipo de desarrollo a fin de desarrollar, validar, y entregar el software de trabajo que responda a las necesidades de sus interlocutores. Las disciplinas son:

1. Modelo. El objetivo de esta disciplina es entender el negocio de la organización, el problema de dominio que se abordan en el proyecto, y determinar una solución viable para resolver el problema de dominio.
2. Aplicación. El objetivo de esta disciplina es transformar su modelo (s) en código ejecutable y realizar un nivel básico de las pruebas, en particular, la unidad de pruebas.
3. Prueba. El objetivo de esta disciplina consiste en realizar una evaluación objetiva para garantizar la calidad. Esto incluye la búsqueda de defectos, validar que el sistema funciona tal como está establecido, y verificando que se cumplan los requisitos.
4. Despliegue. El objetivo de esta disciplina es la prestación y ejecución del sistema y que el mismo este a disposición de los usuarios finales.
5. Gestión de configuración. El objetivo de esta disciplina es la gestión de acceso a herramientas de su proyecto. Esto incluye no sólo el seguimiento de las versiones con el tiempo, sino también el control y gestión del cambio para ellos.

6. Gestión de proyectos. El objetivo de esta disciplina es dirigir las actividades que se lleva a cabo en el proyecto. Esto incluye la gestión de riesgos, la dirección de personas (la asignación de tareas, el seguimiento de los progresos, etc), coordinación con el personal y los sistemas fuera del alcance del proyecto para asegurarse de que es entregado a tiempo y dentro del presupuesto.

7. Entorno. El objetivo de esta disciplina es apoyar el resto de los esfuerzos por garantizar que el proceso sea el adecuado, la orientación (normas y directrices), y herramientas (hardware, software, etc) estén disponibles para el equipo según sea necesario.

10.2.2. PRINCIPIOS AUP

La AUP es ágil, porque está basada en los siguientes principios:

1. El personal sabe lo que está haciendo. La gente no va a leer detallado el proceso de documentación, pero algunos quieren una orientación de alto nivel y / o formación de vez en cuando. La AUP producto proporciona enlaces a muchos de los detalles, si usted está interesado, pero no obliga a aquellos que no lo deseen.

2. Simplicidad. Todo se describe concisamente utilizando un puñado de páginas, no miles de ellos.

11. Agilidad. Ágil ARRIBA El ajuste a los valores y principios de la Alianza Ágil.

4. Centrarse en actividades de alto valor. La atención se centra en las actividades que se ve que son esenciales para el de desarrollo, no todas las actividades que suceden forman parte del proyecto.

5. Herramienta de la independencia. Usted puede usar cualquier conjunto de herramientas que usted desea con el ágil UP. Lo aconsejable es utilizar las herramientas que son las más adecuadas para el trabajo, que a menudo son las herramientas simples o incluso herramientas de código abierto.

6. Adaptación de este producto para satisfacer sus propias necesidades. La AUP producto es de fácil acomodo común a través de cualquier herramienta de edición de HTML. No se necesita comprar una herramienta especial, o tomar un curso, para adaptar la AUP.

11. METODOLOGÍA DE DESARROLLO UWE

UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemática, basada en las técnicas de UML, la notación de UML y los mecanismos de extensión de UML.

Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería web, prestando especial atención en sistematización y personalización (sistemas adaptativos). UWE es una propuesta basada en el proceso unificado y UML pero adaptados a la web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

En el marco de UWE es necesario la definición de un perfil UML (extensión) basado en estereotipos con este perfil se logra la asociación de una semántica distinta a los diagramas del UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las aplicaciones Web. Entre los principales modelos de UWE podemos citar: el modelo lógico-conceptual, modelo navegacional, modelo de presentación, visualización de Escenarios Web y la interacción temporal, entre los diagramas: diagramas de estado, secuencia, colaboración y actividad.

UWE define vistas especiales representadas gráficamente por diagramas en UML.

Además UWE no limita el número de vistas posibles de una aplicación, UML proporciona mecanismos de extensión basados en estereotipos. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son lo que finalmente se utilizarán en las vistas especiales para el modelado de aplicaciones Web. De esta manera, se obtiene una notación UML adecuada a un dominio en específico a la cual se le conoce como Perfil UML.

UWE está especializada en la especificación de aplicaciones adaptativas, y por tanto hace especial hincapié en características de personalización, como es la definición de un modelo de usuario o una etapa de definición de características adaptativas de la navegación en función de las preferencias, conocimiento o tareas de usuario.

Además de estar considerado como una extensión del estándar UML, también se basa en otros estándares como por ejemplo: XMI como modelo de intercambio de formato, MOF para la meta-modelado, los principios de modelado de MDA, el modelo de transformación del lenguaje QVT y XML.

11.1. ACTIVIDADES DE MODELADO UWE

Las actividades base de modelado de UWE son el análisis de requerimientos, el modelo conceptual, el modelo navegacional y el modelo de presentación. A estos modelos se pueden sumar otros modelos como lo son el modelo de interacción y la visualización de Escenarios Web.

El modelo que propone UWE está compuesto por etapas o sub-modelos:

- ✓ Modelo de Casos de Uso
- ✓ Modelo de Contenido
- ✓ Modelo de Usuario
- ✓ Modelo de estructura
- ✓ Modelo Abstracto
- ✓ Modelo de Adaptación
- ✓ modelo de flujo de presentación.
- ✓ modelo de ciclo de vida del objeto.

11.2. MODELO LÓGICO CONCEPTUAL.

UWE apunta a construir un modelo conceptual de una aplicación Web, procura no hacer caso en la medida de lo posible de cuestiones relacionadas con la navegación, y de los aspectos de interacción de la aplicación Web. La construcción de este modelo lógico-conceptual se debe llevar a cabo de acuerdo con los casos de uso que se definen en la especificación de requerimientos. El

modelo conceptual incluye los objetos implicados en las actividades típicas que los usuarios realizarán en la aplicación Web.

11.3. MODELO DE NAVEGACIÓN

Consta de la construcción de dos modelos de navegación, el modelo del espacio de navegación y el modelo de la estructura de navegación. El primero especifica que objetos serán visitados por el navegador a través de la aplicación. El segundo define como se relacionaran.

11.4. MODELO DE PRESENTACIÓN

Describe dónde y cómo los objetos de navegación y accesos primitivos serán presentados al usuario, es decir, una representación esquemática de los objetos visibles al usuario.

11.4.1. INTERACCIÓN TEMPORAL

Presenta los objetos que participan en la interacción y la secuencia de los mensajes enviados entre ellos.

11.4.2. ESCENARIO WEB

Permiten detallar la parte dinámica del modelo de navegación, especificando los eventos que disparan las situaciones, definen condiciones y explícitamente incluyen las acciones que son realizadas. Junto con el modelo de interacción temporal, los escenarios Web proveen la representación funcional dinámica del modelo de navegación.

11.4.3. DIAGRAMAS

Los diagramas usados por UWE, son diagramas UML puro. Entre los más importantes tenemos: Diagramas de estado, de Secuencia, de colaboración y diagramas de Actividad.

11.5. FASES DE LA UWE

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrandose además su atención en aplicaciones personalizadas o adaptativas.

Las fases o etapas a utilizar son:

11.5.1. CAPTURA ANÁLISIS Y ESPECIFICACIÓN DE REQUISITOS:

En simple palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipo de la interfaz de usuario.

11.5.2. DISEÑO DE SISTEMAS:

Se basa en la especificación de requisitos producido por el análisis de los requerimientos (fase de análisis), el diseño define cómo estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

11.5.3. CODIFICACIÓN DE SOFTWARE:

Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior.

11.5.4. PRUEBAS:

Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.

11.5.5. LA INSTALACIÓN O FASE DE IMPLEMENTACIÓN:

Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados, y, eventualmente, configurados; todo ello con el propósito de ser ya utilizados por el usuario final.

Esto incluye la implementación de la arquitectura, de la estructura del hiperespacio, del modelo de usuario, de la interfaz de usuario, de los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones.

11.5.6. EL MANTENIMIENTO:

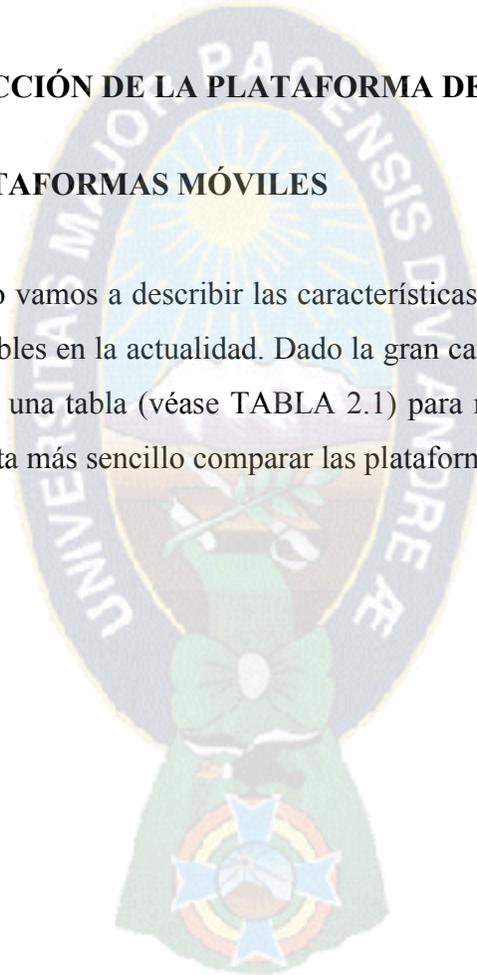
Es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado de la fase de pruebas de control.

12. HERRAMIENTAS A UTILIZAR

12.1. ELECCIÓN DE LA PLATAFORMA DE TRABAJO

12.1.1. PLATAFORMAS MÓVILES

En este apartado vamos a describir las características de las principales plataformas móviles disponibles en la actualidad. Dado la gran cantidad de datos que se indican, hemos utilizado una tabla (véase TABLA 2.1) para representar la información. De esta forma resulta más sencillo comparar las plataformas.



	Apple iOS 7	Android 4.3	Windows Phone 8	BlackBerry OS 7	Symbian 9.5
Compañía	Apple	Open Handset Alliance	Microsoft	RIM	Symbian Foundation
Núcleo del SO	Mac OS X	Linux	Windows NT	Mobile OS	Mobile OS
Licencia de software	Propietaria	Software libre y abierto	Propietaria	Propietaria	Software libre
Año de lanzamiento	2007	2008	2010	2003	1997
Fabricante único	Sí	No	No	Sí	No
Variedad de dispositivos	modelo único	muy alta	media	baja	muy alta
Soporte memoria externa	No	Sí	Sí	Sí	Sí
Motor del navegador web	WebKit	WebKit	Pocket Internet Explorer	WebKit	WebKit
Soporte Flash	No	Sí	No	Si	Sí
HTML5	Sí	Sí	Sí	Sí	No
Tienda de aplicaciones	App Store	Google Play	Windows Marketplace	BlackBerry App World	Ovi Store
Número de aplicaciones	825.000	850.000	160.000	100.000	70.000
Coste publicar	\$99 / año	\$25 una vez	\$99 / año	sin coste	\$1 una vez
Actualizaciones automáticas del S.O.	Sí	depende del fabricante	depende del fabricante	Sí	Sí
Familia CPU soportada	ARM	ARM, MIPS, Power, x86	ARM	ARM	ARM
Máquina virtual	No	Dalvik	.net	Java	No
Aplicaciones nativas	Siempre	Sí	Sí	No	Siempre
Lenguaje de programación	Objective-C, C++	Java, C++	C#, muchos	Java	C++
Plataforma de desarrollo	Mac	Windows, Mac, Linux	Windows	Windows, Mac	Windows, Mac, Linux

TABLA: 2.1: Tabla comparativa de plataformas móviles
FUENTE: (Universidad Pontificia de Valencia - España)

Otro aspecto fundamental a la hora de comparar las plataformas móviles es su cuota de mercado. En la siguiente gráfica podemos ver un estudio realizado por la empresa Gartner Group⁴, donde se muestra la evolución del mercado de los sistemas operativos para móviles según el número de terminales vendidos. Podemos destacar: el importante descenso de ventas de la plataforma Symbian de Nokia; el declive continuo de BlackBerry; como la plataforma de Windows que

⁴ **Gartner Group.** es una empresa consultora y de investigación de las tecnologías de la información con sede en Stamford, Connecticut, Estados Unidos.

parece que no despega; como Apple tiene afianzada una cuota de mercado en torno al 15%. Finalmente destacamos el espectacular ascenso de la plataforma Android, que le ha permitido alcanzar en dos años una cuota de mercado superior al 75%.

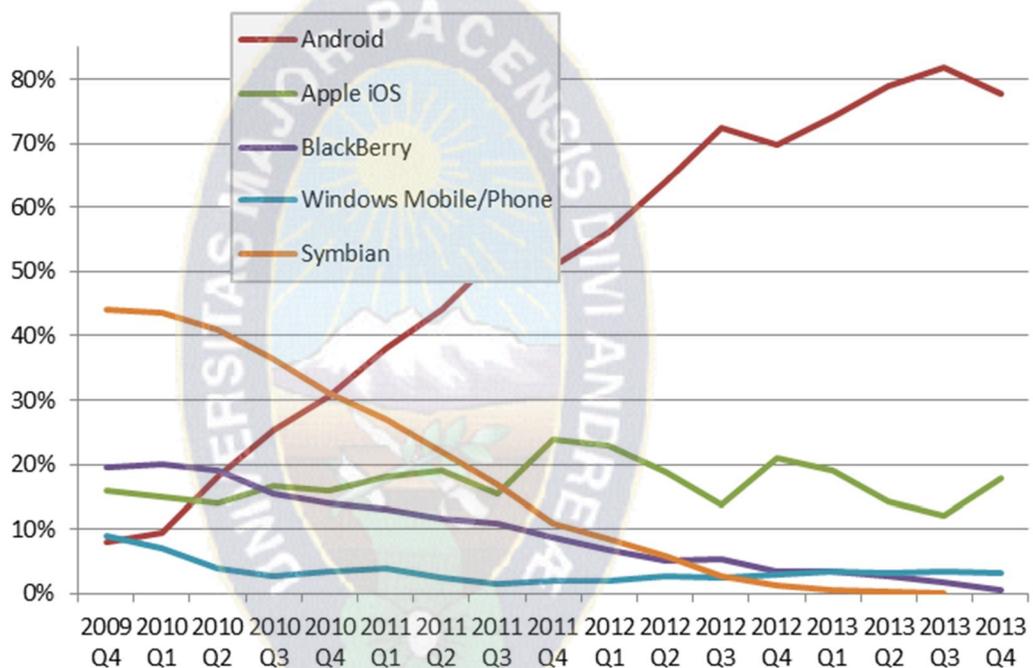


FIGURA 2.5: Porcentaje de teléfonos inteligentes vendidos según su sistema operativo
Hasta el último cuarto del 2013 en el mundo
FUENTE: (Gartner Group)

Para la elección final se ha tenido muy en cuenta la cuota de mercado y el crecimiento de estos sistemas operativos en los últimos años. En la figura anterior (véase FIGURA: 2.5) se puede observar la repartición del mercado de los teléfonos móviles.

Si además tenemos en cuenta la caracterización realizada en apartados anteriores, es fácil desde un inicio descartar un par de plataformas, para las que no están teniendo éxito el desarrollar.

En concreto se descarta BlackBerry y Symbian, ya que no cuentan con una tienda de aplicaciones a la altura de sus competidores y en el caso de Symbian está en pleno proceso de desaparición.

La siguiente plataforma en ser descartada es Windows Phone, se trata de una plataforma joven, tras los intentos de Microsoft de encontrar el sistema operativo perfecto que no pudo conseguir con Windows Mobile. Es probable que se asiente en el mercado y obtenga buenos resultados, pero aún está a mucha distancia de los dos sistemas operativos móviles principales: iOS y Android.

La lucha entre iOS y Android no tiene un vencedor claro y queda en manos del usuario decidir cuál se acerca más a sus preferencias y/o requisitos, en este caso se va a realizar sobre la **plataforma Android**, ya que permite la ejecución de la aplicación en el propio dispositivo sin la necesidad de adquirir una licencia.

En el siguiente apartado, se va a realizar un amplio resumen de la plataforma elegida, así como las herramientas de trabajo utilizadas a lo largo del Trabajo.

12.2. HERRAMIENTAS DE DESARROLLO A USAR

12.2.1. ECLIPSE

El desarrollo en Eclipse con ADT es muy recomendable y es la manera más rápida de comenzar. Con la configuración del proyecto guiada que ofrece, así como la integración de herramientas, editores de XML personalizados, y el panel de resultados de la depuración, ADT le da un impulso increíble en el desarrollo de aplicaciones Android.

Eclipse es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent o Azureus.

Eclipse es también una comunidad de usuarios, extendiendo constantemente las áreas de aplicación cubiertas. Un ejemplo es el recientemente creado Eclipse Modeling Project, cubriendo casi todas las áreas de Model Driven Engineering.

Eclipse fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.

Eclipse fue liberado originalmente bajo la Common Public License, pero después fue re-licenciado bajo la Eclipse Public License. La Free Software Foundation ha dicho que ambas licencias son licencias de software libre, pero son incompatibles con Licencia pública general de GNU (GNU GPL).

12.2.2. ANDROID DEVELOPMENT TOOLS (ADT)

Android Development Tools (ADT) es un plugin para el IDE Eclipse que está diseñado para darle un poderoso entorno integrado en el que la construcción de aplicaciones de Android.

ADT amplía las capacidades de Eclipse para permitirle configurar rápidamente nuevos proyectos para Android, crear una interfaz de usuario de la aplicación, agregue los paquetes basados en la API de Android Framework, depurar sus aplicaciones utilizando las herramientas del SDK de Android, e incluso exportar firmado (o signo) .apk archivos con el fin de distribuir la aplicación.

12.3. GPS

Sistema de Posicionamiento Global (Global Positioning System) es un sistema de navegación basado en satélites y está integrado por 24 satélites puestos en órbita por el Departamento de defensa de los Estados Unidos. Originalmente, fue pensado para aplicaciones militares, aunque a partir de los años 80's el gobierno de USA puso el sistema de navegación disponible a la población civil. El GPS funciona en cualquier condición climatológica, en cualquier parte del mundo las 24 horas del día. No hay ningún costo de suscripción o cargos iniciales de preparación para usar el GPS.

12.3.1. CARACTERÍSTICAS Y USOS

Los satélites GPS (24 en operación permanente y 3 de respaldo) giran alrededor de la tierra dando dos vueltas completas al día dentro de una órbita muy precisa transmitiendo señales a la tierra que indican su ubicación y la hora que les proporciona un reloj atómico que traen a bordo. Todos los satélites están sincronizados, de tal manera que las señales transmitidas se efectúan en el mismo instante. Los receptores de GPS reciben esta información y la utilizan para

triangular y calcular la localización exacta del receptor. Las señales se mueven a la velocidad de la luz y llegan a los receptores en diferentes tiempos ya que algunos satélites están más alejados que otros. Esencialmente, el receptor GPS en la tierra compara el tiempo que toma una señal que se transmite desde uno de los satélites en el espacio, con el tiempo en que esta misma señal es recibida por el receptor.

La diferencia en tiempo, le indica al receptor de GPS que tan lejos está del satélite. Ahora bien, con la obtención de varias lecturas de diferentes satélites, el receptor de GPS puede determinar con gran exactitud la posición del usuario y desplegarla en un mapa electrónico en la unidad receptora. Cuando el receptor estima la distancia de al menos cuatro satélites GPS, puede calcular su posición en tres dimensiones, Longitud, Latitud y Altitud.

Los receptores de GPS en la tierra, calculan rápidamente y con gran precisión los tres valores necesarios para ubicar en un mapa electrónico a un usuario. El valor de Longitud (Meridianos) que es la referencia con respecto al meridiano de GREENWICH o PRIME MERIDIAN hacia el Este (E) o hacia el Oeste (W), el valor de Latitud (Paralelos) que es la referencia con respecto al ECUADOR, hacia el Norte (N) o hacia el Sur (S) y el valor de Altitud que es la referencia con respecto al nivel medio del mar.

12.4. PLATAFORMA DE TRABAJO

12.4.1. ANDROID

Android es un sistema operativo basado en el kernel 2.6 de Linux diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes o tabletas, inicialmente desarrollado por Android Inc⁵. Empresa a la que Google respaldó económicamente y que en 2005 la compró.

La compañía original era de Palo Alto, California (fundada en 2003). Entre los cofundadores de Android que se fueron a trabajar a Google están Andy Rubin (co-fundador de Danger), Rich Miner (co-fundador de Wildfire Communications, Inc.), Nick Sears (alguna vez VP en T-Mobile), y Chris White (quien encabezó el diseño y el desarrollo de la interfaz en WebTV). Android se presentó en el 2007 junto con la

Open Handset Alliance, un consorcio de compañías de hardware, software y telecomunicaciones, con la intención de avanzar en los estándares de los sistemas abiertos. El primer teléfono con Android fue el HTC Dream⁶(véase FIGURA

2.6), que empezó a venderse en octubre del 2008.



FIGURA 2.6: El HTC DREAM primer teléfono en incorporar Android
FUENTE: (Wikipedia marzo 2014)

12.4.2. OPEN HANDSET ALLIANCE

⁵ **Android Inc.** fue fundada en Octubre de 2003 por **Andy Rubin**, **Rich Miner**, Nick Sears y Chris White. Sus intenciones iniciales eran las de desarrollar un sistema operativo inteligente.

⁶ El **HTC Dream** (comercializado también como **T-Mobile G1** y denominado popularmente Google Phone o GPhone) es un dispositivo de telefonía móvil construido por **HTC**. Fue lanzado al mercado el 22 de octubre de 2008, a un costo estimado de 179 dólares U.S.A

El 5 de noviembre de 2007 la Open Handset Alliance, un consorcio de varias compañías entre las que están Texas Instruments, Broadcom Corporation, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, Intel, LG, Marvell Technology Group, Motorola, y T-Mobile; se estrenó con el fin de desarrollar estándares abiertos para dispositivos móviles. Junto con la formación de la Open Handset Alliance, la OHA estrenó su primer producto, Android, una plataforma para dispositivos móviles construida sobre la versión 2.6 de Linux.

El 9 de diciembre de 2008, se anunció que 14 nuevos miembros se unirían al proyecto Android, incluyendo PacketVideo, ARM Holdings, Atheros Communications, Asustek, Garmin, Softbank, Sony Ericsson, Toshiba, Vodafone y ZTE

12.4.3. HISTORIA DE ACTUALIZACIONES

El historial de versiones del sistema operativo Android se inició con el lanzamiento de Android beta en noviembre de 2007. La primera versión comercial, Android 1.0, fue lanzada en septiembre de 2008. Android es un sistema operativo móvil desarrollado por Google y la Open Handset Alliance, y ha visto un número de actualizaciones a su sistema operativo base desde su lanzamiento original. Estas actualizaciones típicamente corrigen fallos de programa y agregan nuevas funcionalidades. Desde abril de 2009, Las versiones de Android han sido desarrolladas bajo un nombre en clave y lanzamiento en orden alfabético: Apple Pie, Banana Bread, Cupcake, Donut, Éclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwich, Jelly Bean y KitKat (el nombre de una chocolatina de Nestle). La actualización más reciente es kitkat 4.4 la cual fue anunciada en septiembre de 2013, y lanzada en el proyecto Android Open Source un mes después.

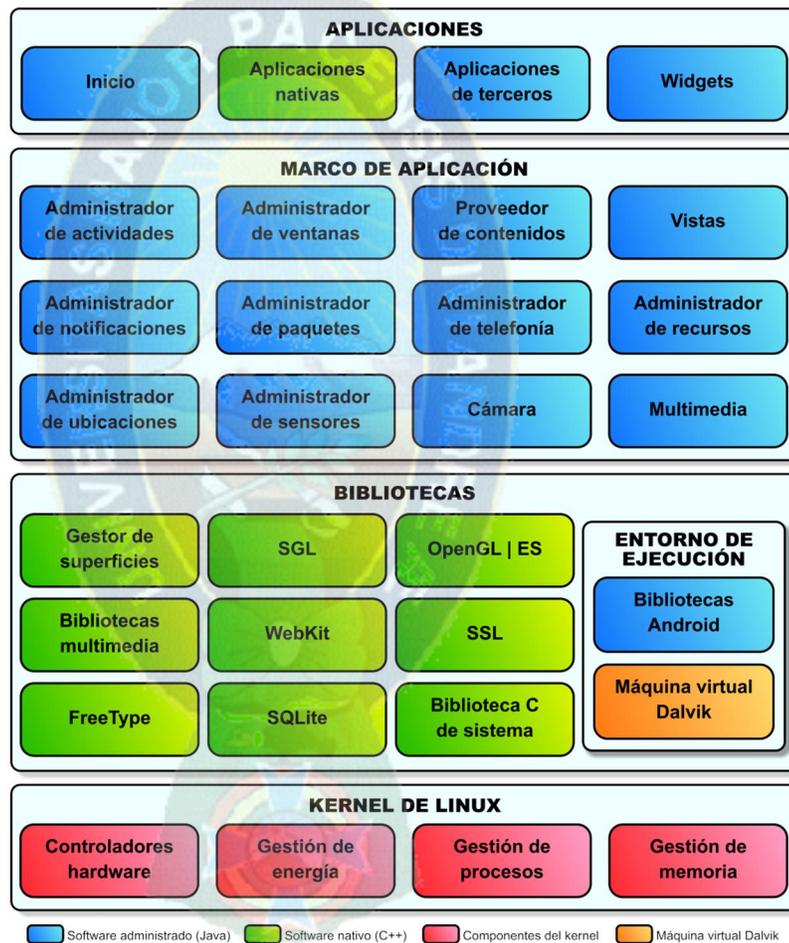
12.4.4. VERSIONES



FIGURA 2.7: Evolución de las versiones de Android
FUENTE: (<http://www.android.com/> 2014)

12.4.5. ARQUITECTURA DE ANDROID

A continuación se muestra la arquitectura de Android, consta de cinco módulos principales: El kernel de Linux, bibliotecas, entorno de ejecución, marco de aplicación y aplicaciones.



A continuación se va proceder a detallar los principales componentes de Android:

12.4.5.1. KERNEL DE LINUX:

El núcleo del sistema operativo Android es un kernel Linux versión 2.6, se trata de una adaptación similar al que puede incluir en cualquier distribución de Linux. Nos proporciona una capa de abstracción lo que permite que se pueda acceder a esos componentes del teléfono, sus características o el modelo. Además otras funciones del kernel son: gestionar los diferentes recursos del teléfono (energía, memoria, etc.) y del sistema operativo en sí: procesos, elementos de comunicación, etc.

12.4.5.2. BIBLIOTECAS

FIGURA 2.8: Arquitectura de Android
FUENTE: (<http://www.android.com/> 2014)

La capa que se sitúa justo sobre el kernel la componen las bibliotecas nativas de Android. Estas bibliotecas están escritas en C o C++. Su función es la de proporcionar funcionalidad a las aplicaciones, garantiza una mayor eficiencia al evitar tener que codificar cada vez tareas que se repiten con cierta frecuencia.

12.4.5.3. ENTORNO DE EJECUCION

En realidad no puede considerarse una capa como tal, ya que también está formado por bibliotecas. Su principal componente es la máquina virtual Dalvik, la cual ejecuta las aplicaciones no nativas de Android. Éstas son compiladas en un formato específico para la máquina virtual Dalvik, lo que nos asegura que la aplicación se podrá ejecutar en cualquier dispositivo Android que cumpla con los requisitos de mínima versión. Durante el proceso de compilación se generan los habituales archivos .class, pero esos archivos son convertidos a archivos .dex (Dalvik executable). Se trata de una medida de optimización, ya que los archivos .dex ocupan hasta un 50% menos lo que permite ahorrar espacio y acelerar los procesos de carga.

12.4.5.4. MARCO DE APLICACIÓN:

Formada por las clases y servicios que utilizan directamente las aplicaciones para realizar sus funciones, siendo la mayoría de sus componentes bibliotecas Java que acceden a los recursos a través de la máquina virtual Dalvik.

12.4.5.5. APLICACIONES:

Se encuentran las aplicaciones del dispositivo. Todas utilizan el mismo marco de aplicación para acceder a los servicios proporcionados por el sistema operativo, lo que quiere decir que se pueden desarrollar aplicaciones que usen los mismos recursos que usan las aplicaciones nativas, siendo una de las mayores virtudes de Android: control total por parte del usuario del software que se ejecuta en su teléfono.

12.4.5.6. ESTRUCTURA DE UNA APLICACIÓN ANDROID

Los componentes básicos de una aplicación Android son: Activities, intents, vistas, servicios, proveedor de contenido, receptores de difusión, manifiesto.

12.4.5.7. ACTIVITIES:

Son los componentes de la interfaz que corresponden a una pantalla. Se pueden visualizar como un mazo de cartas en el que se tienen varias cartas pero solo una está arriba. Una aplicación para una lista de cosas para hacer puede tener una actividad para ingresar las cosas por hacer y otra actividad para mostrar un listado, en conjunto estas actividades forman una aplicación.

12.4.5.8. INTENTS

Son mensajes que provocan notificaciones o cambios de estatus, que al ser recibidos por actividades o servicios pueden levantar procesos. De esta forma se unen componentes dentro de la misma aplicación o de diferentes aplicaciones.

12.4.5.9. VIEWS:

Son los componentes de la interfaz de usuario, diferentes vistas pueden agruparse a través de grupos logrando una jerarquía, esto se logra a través de la disposición de los componentes a través de un archivo XML.

12.4.5.10. SERVICES:

Son componentes que ejecutan operaciones en segundo plano y no tienen una interfaz de usuario. Por ejemplo, cuando escuchamos música, hay un servicio

encargado de la reproducción que se ejecuta de fondo y la aplicación que manipulamos le manda mensajes a este servicio diciéndole que se detenga, pause o reproduzca la siguiente canción.

12.4.5.11. CONTENT PROVIDERS:

Content Providers: representan la abstracción para almacenar y obtener datos permanentes y aplicaciones diferentes. El sistema incluye algunos proveedores de contenidos útiles (audio, video, etc) y además pueden desarrollarse nuevos.



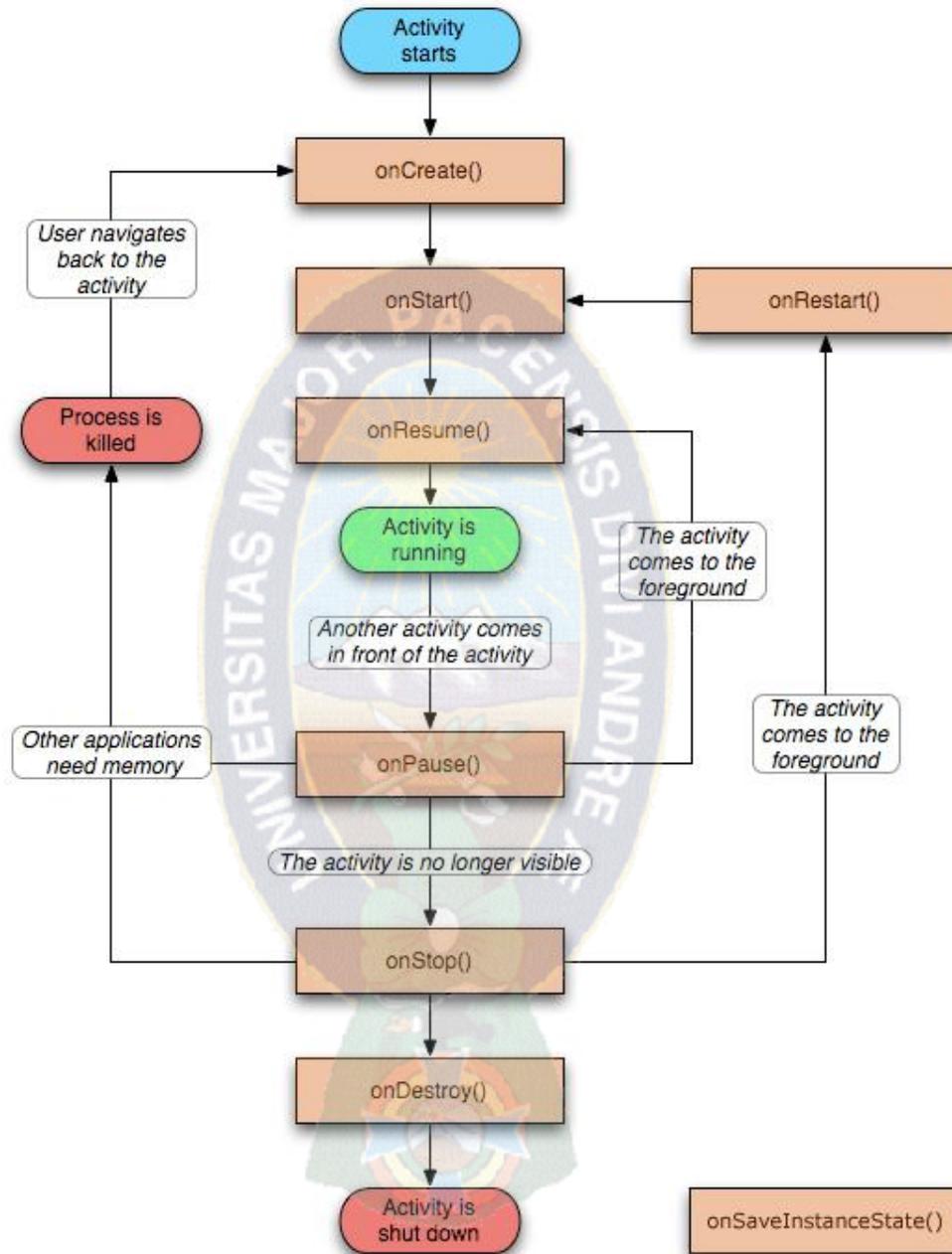


FIGURA 2.9: Estructura de una Aplicación Móvil Android
 FUENTE: (<http://www.android.com/>)

12.4.5.12. MANIFEST:

El archivo AndroidManifest.xml es donde se configura la aplicación, se agregan actividades, se asignan permisos, etc.

12.4.5.13. BROADCAST RECEIVERS:

Son componentes que responden a avisos y anuncios de difusión (broadcast). Estos avisos provienen del sistema (batería baja, una llamada entrante, etc.) y de las aplicaciones (pasando avisos de una aplicación a otra). Aunque no muestran una interfaz de usuarios algunas veces utilizan barras de progreso para mostrar avances. Estos se activan a través de mensajes asíncronos llamados “intents”.

12.5. GOOGLE MAPS

Google Maps es un servidor de aplicaciones de mapas en la web que pertenece a Google. Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle Google Street View. Desde el 6 de octubre de 2005, Google Maps es parte de Google Local.

Permite a los usuarios explorar el mundo con mapas ricos proporcionadas por Google. Identificar lugares con marcadores personalizados, aumentar los datos de los mapas con las superposiciones de imágenes,

12.6. OTRAS HERRAMIENTAS

En cuanto a las herramientas de desarrollo del proyecto.

- Plataforma: Microsoft Windows 7 por encontrarse este sistema operativo instalado en los equipos de computación de la empresa, siendo también compatible con Microsoft Windows XP y Linux.
- PHP (acrónimo de “PHP: Hypertext Preprocessor”) es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutando en el servidor. Páginas dinámicas
- *JavaScript*: que está incrustado en todos los navegadores actuales y puede mezclarse su código en documentos HTML, para notificaciones y validación de datos, el lenguaje *script*.
- CSS. Hojas de Estilo en Cascada (*Cascading Style Sheets*), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla. Se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación.
- MySQL: Gestor de Base de datos el cual administra nuestra base de datos, y al mismo tiempo interactúa con el lenguaje.
- Adobe Dreamweaver CSS: se utilizará esta aplicación de diseño Web para minimizar esfuerzo y tiempo.
- Adobe Flash CSS Professional: se empleará para darle un interfaz de usuario amigable e intuitivo.
- Adobe Photoshop CSS: aplicación que permitirá el diseño gráfico con el cual se elaborarán las distintas apariencias del sistema.
- Microsoft Visio 2003 : Para Realizar los diseños UML y demás diagramas.



CAPITULO III

Análisis y Diseño

13.1. INTRODUCCIÓN

En este capítulo se dará a conocer todo el procedimiento a seguir para lograr los objetivos descritos en el capítulo I. Se plantea la solución, teniendo en cuenta ciertas restricciones, finalmente el diseño de la aplicación.

Se describirá el proceso de las metodologías AUP (Proceso Unificado Ágil) con:

- ✓ Sus respectivas disciplinas como ser: Modelado, Implementación, Prueba
- ✓ Sus diferentes etapas como ser la iniciación, elaboración y construcción de la Aplicación de Localización y control de dispositivos Smartphone Android.

Con el modelado y desarrollo que nos permite el modelo UWE (Ingeniería Web Basada en UML), la metodología nos permite:

- ✓ El análisis de los requisitos (definición de actores, casos de uso por actores y especificaciones de cada caso).
- ✓ Capa conceptual, para el contenido (modelo del Dominio)
- ✓ Capa de Navegación, que incluye modelos estáticos y dinámicos.
- ✓ Capa de Presentación, modelo de flujo de presentación
- ✓ Modelo abstracto de interfaz de usuario.

13.2. FASE DE COMIENZO

En la fase de la iniciación el principal objetivo es registrar los requerimientos de los interesados

13.2.1. DISCIPLINA DEL MODELADO

El objetivo de esta disciplina es entender el tipo de aplicación que se quiere diseñar, saber exactamente el dominio del problema, que aborda al proyecto identificando una solución viable para hacer frente al problema.

13.2.1.1. ANÁLISIS DE REQUISITOS

Nos Ayudara a la captura de los requisitos finales para la elaboración de la aplicación móvil.

d. DEFINIR ACTORES

En la Tabla 3.1 se describe a los actores y sus descripciones involucrados en el manejo y administración de la aplicación.

ACTOR	DESCRIPCIÓN
 ADMINSTRADOR	Es la persona que tendrá la total administración de la aplicación
 e. TERMINAL MOVIL E	Corresponde a la terminal móvil a ser localizada y controlada

F

INIR

Tabla 3.1: Definición de Actores.
Fuente: [Elaboración Propia]

RELACIONES ENTRE ACTORES



f. ANÁLISIS DE CASO DE USO

Siguiendo el Proceso de Desarrollo de Software Unificado aplicado a la ingeniería Web, se propone los casos de uso necesarios para capturar los requisitos de la aplicación. (Ver figura 3.1)

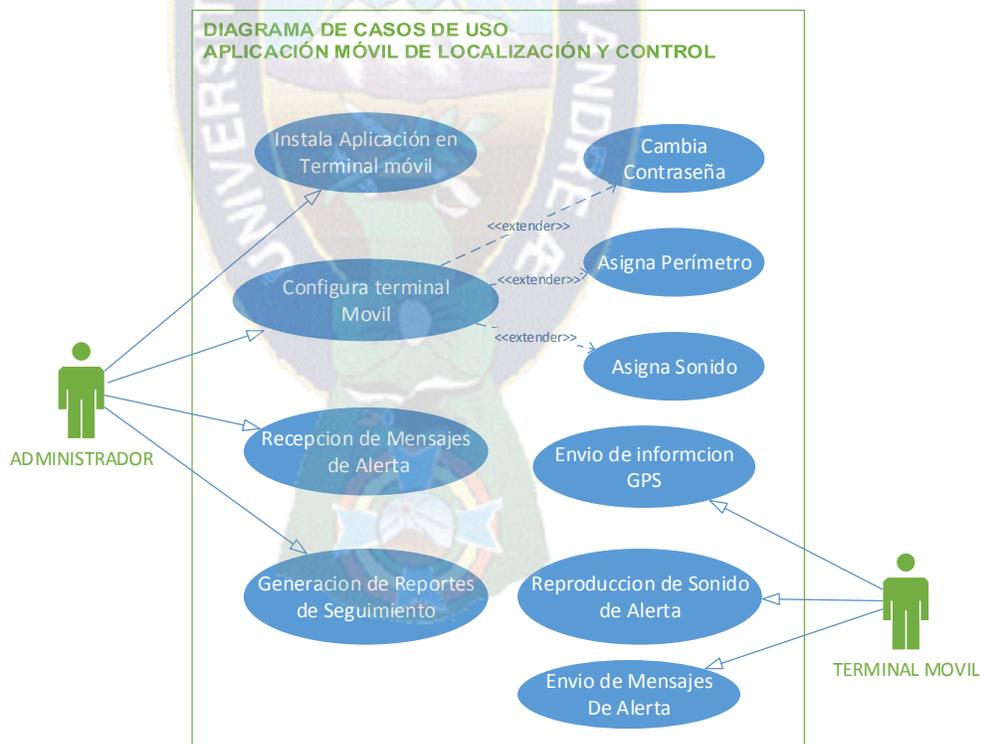


Figura 3.1: Análisis de Caso de Uso
Fuente: [Elaboración Propia]

13.2.2. DISCIPLINA DE IMPLEMENTACIÓN

13.2.2.1.CASO DE USO POR ACTOR (ACTOR ADMINISTRADOR)

En la figura 3.2 se muestra el caso de uso del administrador.

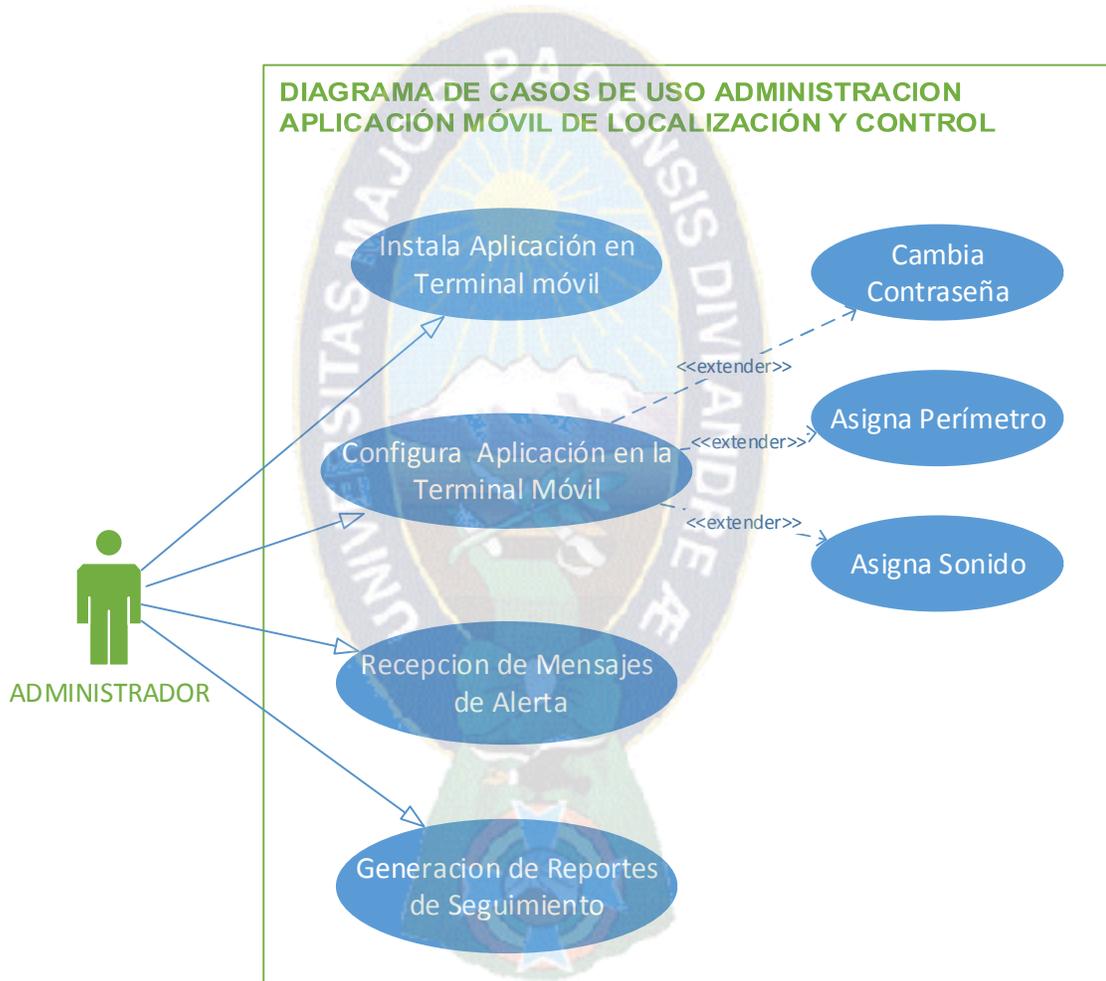


Figura 3.2: Análisis de Caso de Uso Administrador
Fuente: [Elaboración Propia]

a. ESPECIFICACIÓN DE CASO DE USO DEL ADMINISTRADOR

Nombre	Instalar Aplicación en Terminal Móvil	
Actor	Administrador	
Propósito	Crear una aplicación Android, que posteriormente será instalado por el administrador a las terminales móviles a ser localizadas y controladas.	
Resumen	Instalar la aplicación de localización y control en dispositivos Android	
Flujos Principal	Evento Actor	Evento Sistema
	1. El Administrador ingresa Contraseña de Seguridad para posterior instalación de la aplicación	1. El sistema hace verificación de permisos y compatibilidad de la aplicación.
Flujos Alternativos	No existen	
Precondición	La Terminal móvil donde se instalar la aplicación deberá ser Android y deberá cumplir las especificaciones y requerimientos mínimos para su instalación.	
Pos Condición	El Administrador deberá terminar sesión y guardar cambios	

Nombre	Configurar la Aplicación en la Terminal Móvil
Actor	Administrador

Propósito	Proporcionar un entorno de configuración para la terminal móvil a ser localizada y controlada, en la que encontraremos opciones como ser: la asignación de perímetros, opciones de conexión con el servidor, configuración de mensajes, configuración de sonidos.	
Resumen	Configurar las aplicación en la terminal móvil a ser localizada y controlada	
Flujos Principal	Evento Actor	Evento Sistema
	1. El Administrador ingresa Contraseña de Seguridad	1. Se configuran parámetros de asignación para el funcionamiento de la aplicación
Flujos Alternativos	No existen	
Precondición	La Aplicación deberá estar instalada en la terminal móvil y el administrador deberá contar con el código de ingreso	
Pos Condición	El Administrador deberá terminar sesión y guardar cambios	

Tabla 3.3 Configuración de la aplicación móvil.
Fuente: [Elaboración Propia]

Nombre	Recepción de Mensajes de Alerta
Actor	Administrador

Propósito	Tiene el propósito de alertar al administrador con un mensaje que existe una terminal que se salió del perímetro de control y genera automática mente un registro.	
Resumen	Alerta al administrador cuando una terminal móvil sale del perímetro de asignación.	
Flujos Principal	Evento Actor	Evento Sistema
	1. El Administrador es alertado con un mensaje de texto.	1. La Aplicación verifica el perímetro de control asignado a la terminal movil, y almacena un registro en la base de datos si existiera alguna anomalía.
Flujos Alternativos	La Aplicación funciona en segundo plano	
Precondición	La Aplicación deberá estar instalada en la terminal móvil.	
Pos Condición	ninguna	

Tabla 3.4 Configuración de la aplicación móvil.
Fuente: [Elaboración Propia]

Nombre	Generación de Reportes de Seguimiento
Actor	Administrador

Propósito	Tiene el propósito generar reportes, diarios semanales y mensuales de faltas cometidas por las terminales móviles por salirse de perímetros de control.	
Resumen	Generación de Reportes	
Flujos Principal	Evento Actor	Evento Sistema
	<ol style="list-style-type: none"> 1. El Administrador ingresa código de ingreso 2. Se dirige a la opción de generación de reportes 3. Imprime reporte 	<ol style="list-style-type: none"> 1. Se verifica contraseña 2. Elige parámetros de generación de reporte. 3. General el Reporte 4. Genera la Opción de Impresión
Flujos Alternativos	La Aplicación funciona en segundo plano	
Precondición	La Aplicación deberá estar instalada en la terminal móvil.	
Pos Condición	ninguna	

Tabla 3.5 Configuración de la aplicación móvil.
Fuente: [Elaboración Propia]

13.2.2.2.CASO DE USO POR ACTOR (ACTOR TERMINAL MÓVIL)

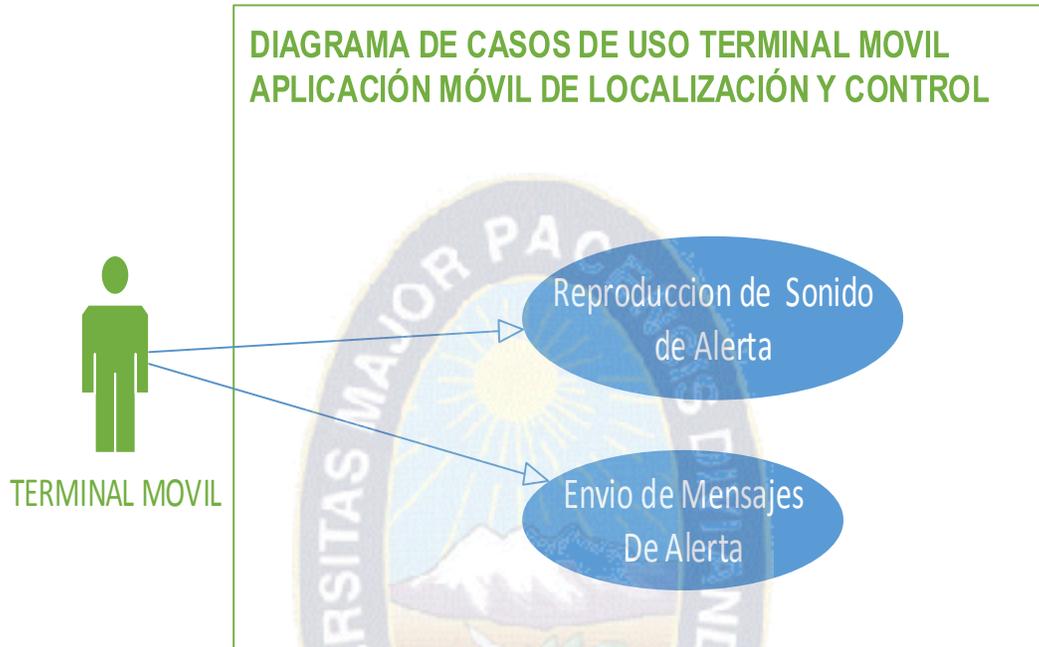


Figura 3.3: Análisis de Caso de Uso (Actor Terminal Móvil)
Fuente: [Elaboración Propia]

b. ESPECIFICACIÓN DE CASO DE USO TERMINAL MÓVIL

Nombre	Reproducción de Sonido de Alerta	
Actor	Terminal Móvil	
Propósito	Tiene el propósito de alertar, producir un sonido para que la persona que porte la terminal móvil sepa que está a punto de pasar el perímetro de asignación y control.	
Resumen	Alerta a la terminal móvil si está a punto de salir del perímetro de asignación y control.	
Flujos Principal	Evento Actor	Evento Sistema
	1. Está a punto de salirse del perímetro de asignación y control.	1. La Aplicación verifica el perímetro de control asignado a la terminal móvil, y si está a punto de salirse emite un sonido de alerta.
Flujos Alternativos	ninguna	
Precondición	La Aplicación deberá estar instalada en la terminal móvil y debe salirse del perímetro de asignación y control.	
Pos Condición	ninguna	

Tabla 3.6 Configuración de la aplicación móvil.

Fuente: [Elaboración Propia]

Nombre	
Actor	Terminal Movil

Propósito	Tiene el propósito de enviar un mensaje de alerta al administrador cuando la terminal móvil se ha salido del perímetro de asignación y control.	
Resumen	Envía Mensajes de Alerta al Administrador	
Flujos Principal	Evento Actor	Evento Sistema
	1. La Terminal móvil se sale del perímetro de asignación y control.	1. La Aplicación verifica que la terminal móvil se salió del perímetro de control asignado, entonces envía un mensaje de alerta al administrador. 2. Genera un registro y se almacena en la base de datos.
Flujos Alternativos	ninguna	
Precondición	La Aplicación deberá estar instalada en la terminal móvil.	
Pos Condición	ninguna	

Tabla 3.7 Configuración de la aplicación móvil.
Fuente: [Elaboración Propia]

13.3. FASE DE ELABORACIÓN

Esta fase tiene como objetivo probar la arquitectura de la aplicación, hacer un prototipo de arquitectura que elimine los riesgos técnicos para probar que el proyecto es factible.

13.3.1. DISCIPLINA DE MODELADO

13.3.1.1. CAPA CONCEPTUAL

Está representado por el Diseño Conceptual, Diagrama Entidad Relación y Diagrama Físico.

g) DISEÑO CONCEPTUAL



Figura 3.3: Diseño Conceptual
Fuente: [Elaboración Propia]

h) D

DIAGRAMA ENTIDAD RELACIÓN

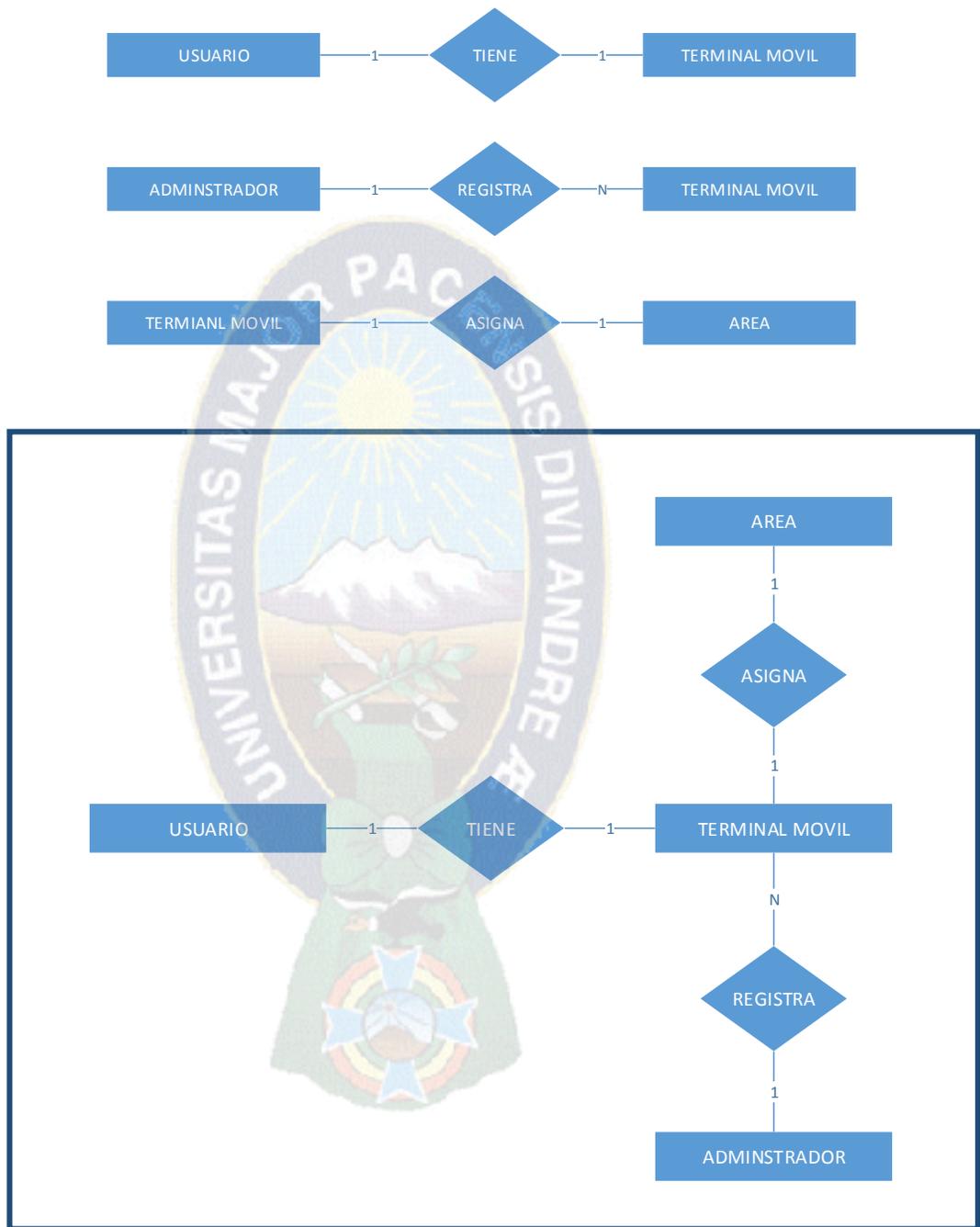


Figura 3.4: Diagrama Entidad Relación
Fuente: [Elaboración Propia]

i) M

ODELO FÍSICO

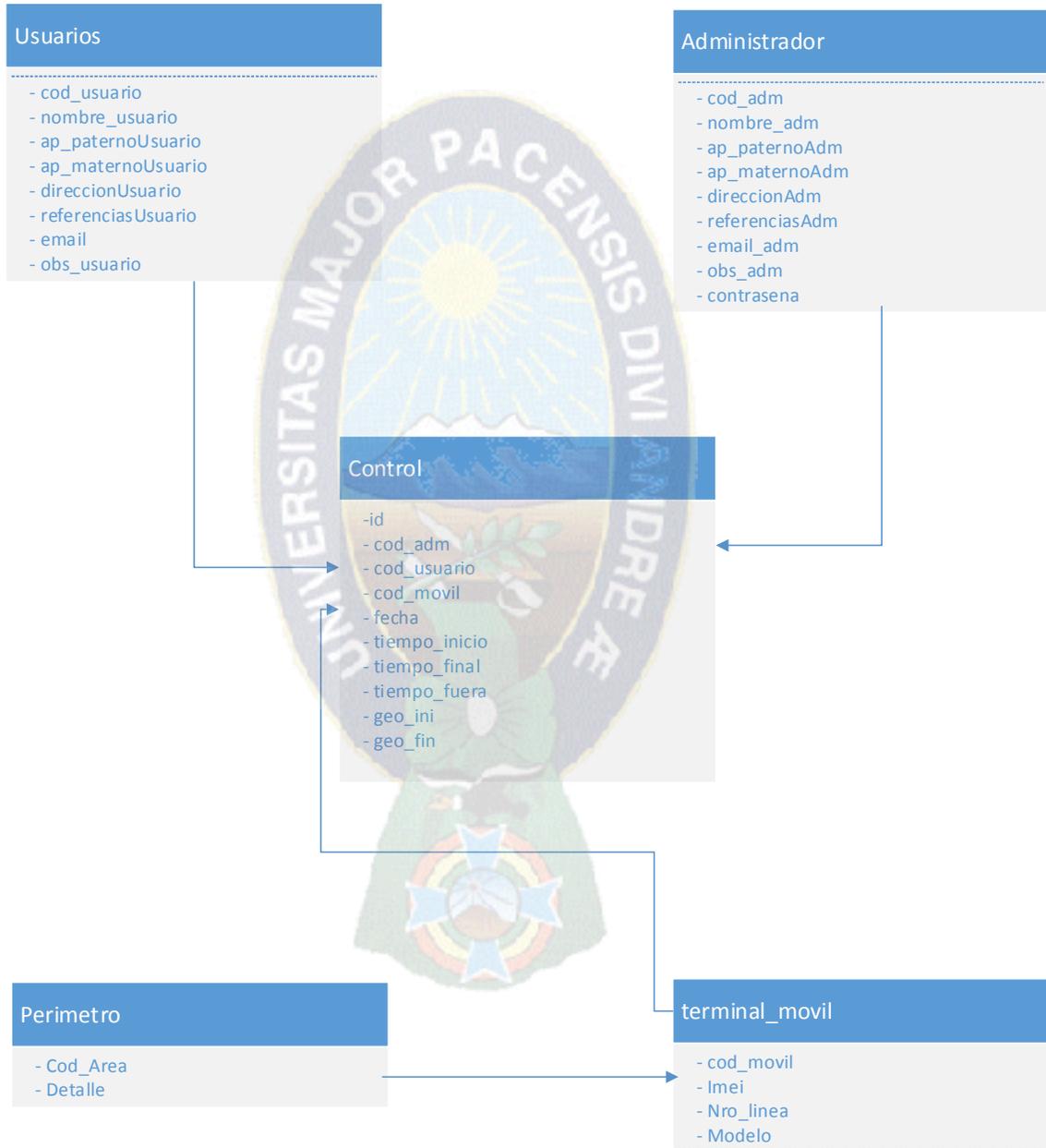


Figura 3.5: Diseño Modelo Físico
Fuente: [Elaboración Propia]

j) DISEÑO DE NAVEGACIÓN

Www.entubusca.com

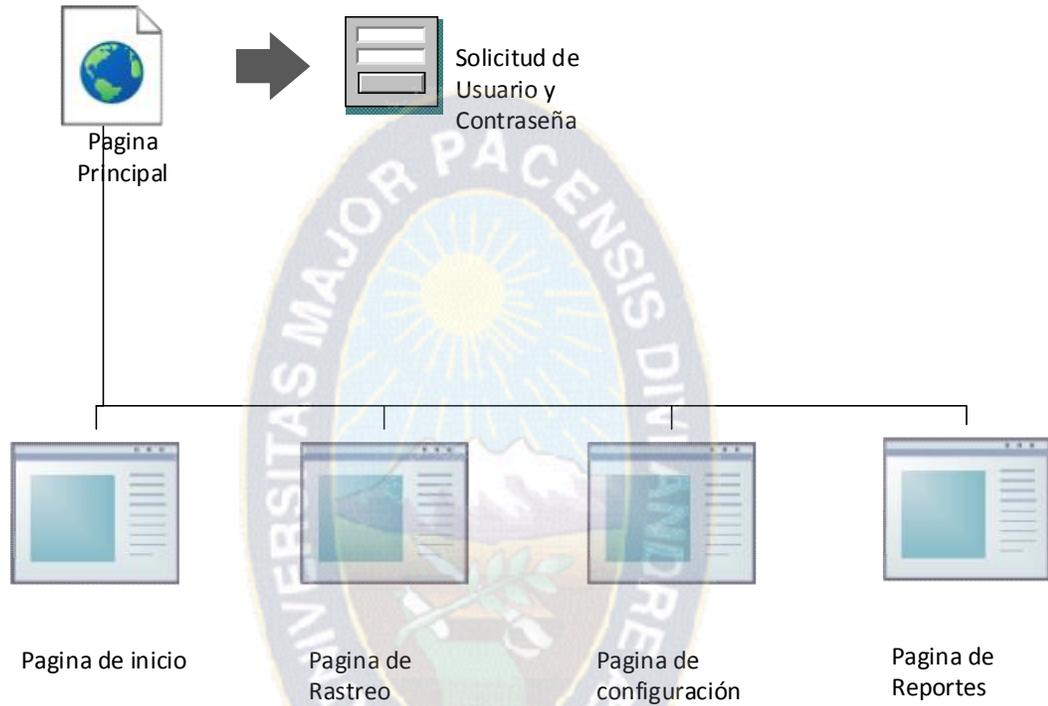


Figura 3.6: Diseño De Navegación
Fuente: [Elaboración Propia]

k) DIAGRAMA GENERAL DEL FUNCIONAMIENTO DEL SISTEMA COMPLETO

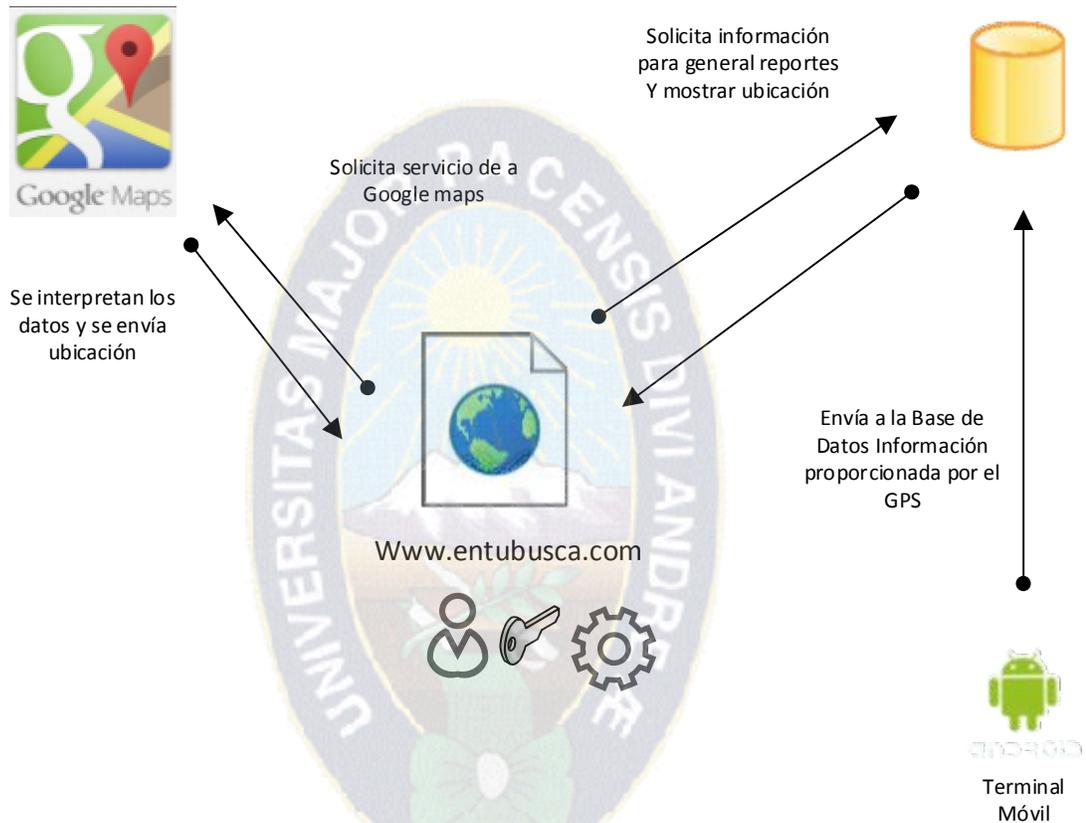


Figura 3.7: Diseño De Funcionamiento del Sistema Completo
Fuente: [Elaboración Propia]

I) DISEÑO DE LA INTERFACE



The image shows a login form titled "EN TU BUSCA" with a sub-header "INGRESAR". The form includes a "Usuario" field with the placeholder text "Ingrese nombre de usuario", a "Contraseña" field with a masked password "*****", an "ENTRAR" button, and two links: "Olvide mi Contraseña" and "Registrar terminal". The background features a large, semi-transparent watermark of the University of Pinar del Rio logo, which includes a sun, mountains, and a cross.

Figura 3.8: Ingresando como administrador
Página principal: [www.entubusca.com]
Fuente: [Elaboración Propia]

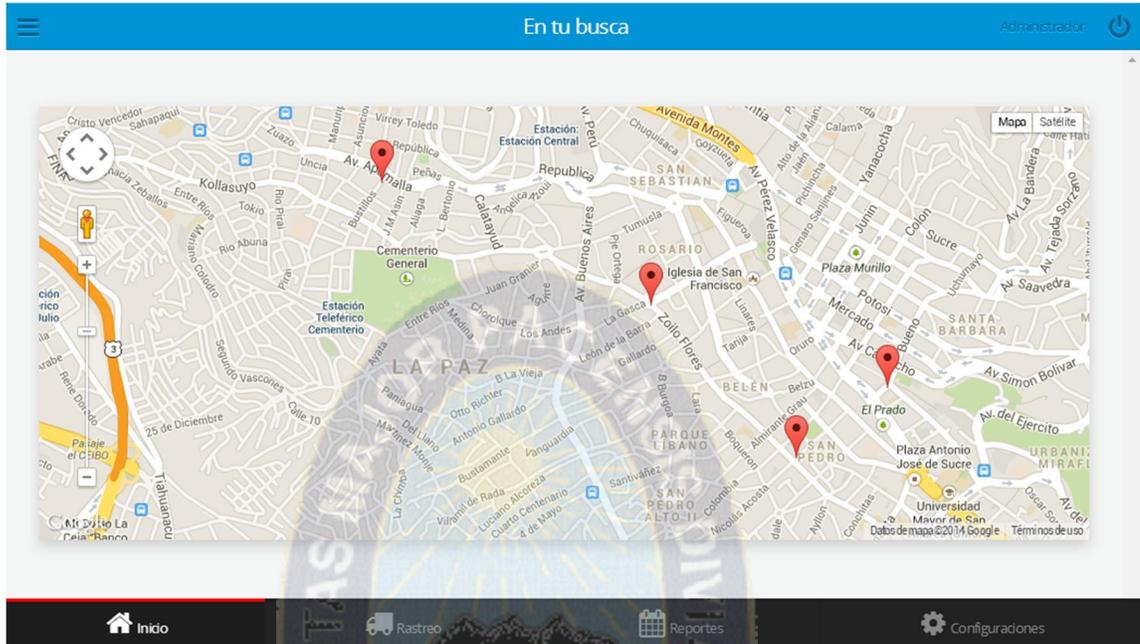


Figura 3.9: Pantalla de Inicio
Fuente: [Elaboración Propia]

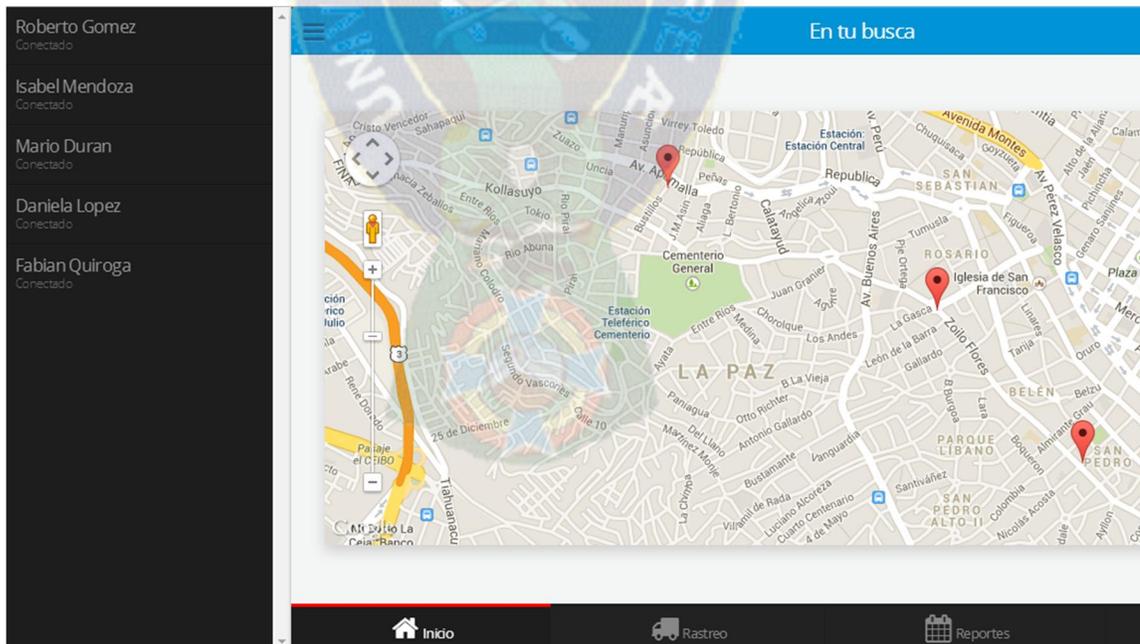


Figura 4.0: Pantalla de Inicio
Fuente: [Elaboración Propia]

En tu busca Administrador

Usuario: Roberto Gomez Desde: 04/05/2014 00:00 Hasta: 04/05/2014 00:00 Generar reporte

Fecha	Hora inicio	Hora fin	Infracción	Tiempo fuera	LatLong inicio	LatLong fin
05/06/2014	09:00:00	12:00:00	No	00:00:00	maps.google.com/maps?q=-16.5339457,-68.0967228	maps.google.com/maps?q=-16.13325436,-68.7685839
07/06/2014	13:30:00	18:40:00	Si	45:40:10	maps.google.com/maps?q=-16.2534658,-68.5554536	maps.google.com/maps?q=-16.0096746,-68.9978697
09/06/2014	10:00:00	12:00:00	Si	00:10:50	maps.google.com/maps?q=-16.6557778,-68.5739164	maps.google.com/maps?q=-16.5453680,-68.6758576
12/06/2014	08:00:00	12:30:00	No	00:00:00	maps.google.com/maps?q=-16.9985205,-68.5666741	maps.google.com/maps?q=-16.45365890,-68.56645341

Inicio Rastreo Reportes Configuraciones

Figura 4.1: Pantalla Genera Reporte
Fuente: [Elaboración Propia]

Usuario: Daniela Lopez Desde: 2014-06-11 14:45:00 Hasta:

Inicio

Figura 4.2: Rastreo de Terminales Móviles
Fuente: [Elaboración Propia]

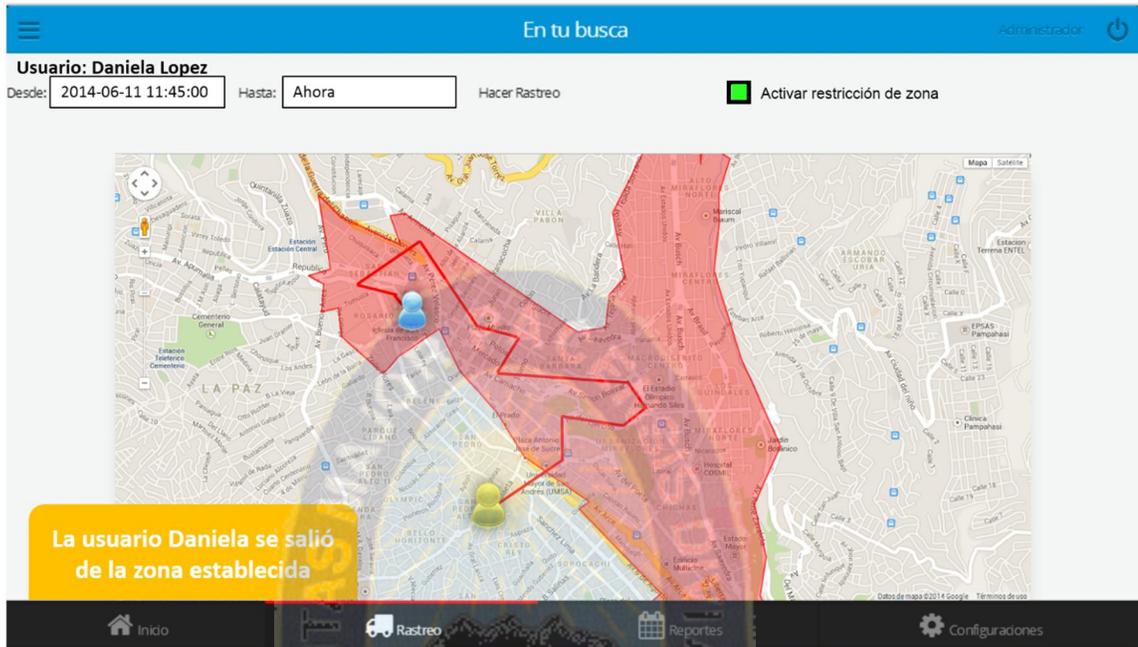


Figura 4.3: Rastreo de terminales Móviles mas restricción de zona
Fuente: [Elaboración Propia]

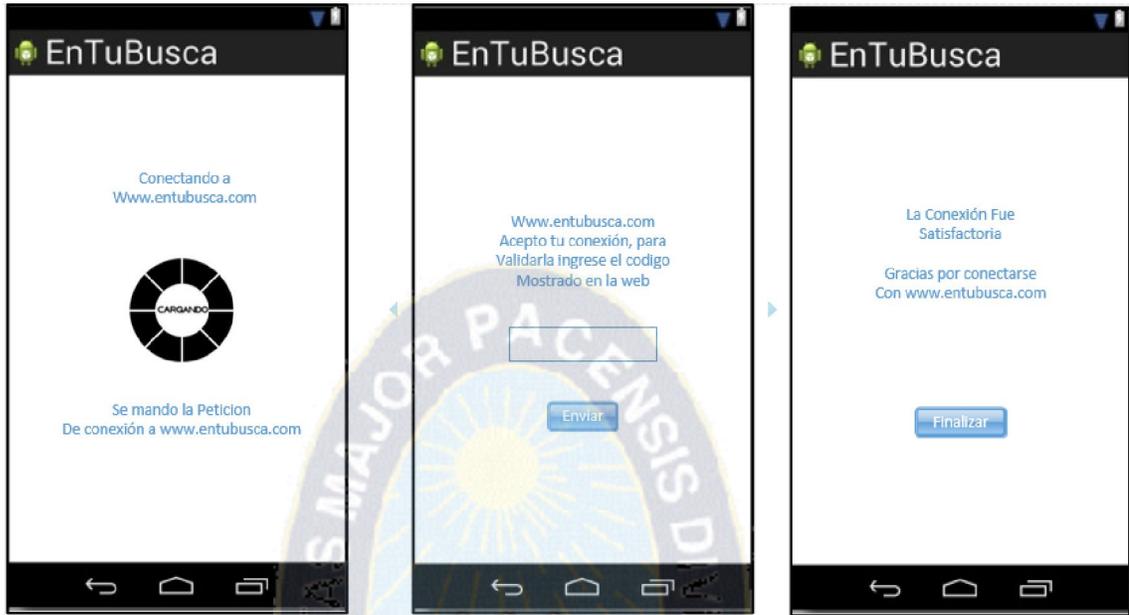


Figura 3

Figura 4.4: Registro de Terminales Móviles
Fuente: [Elaboración Propia]



Figura 4.5: Respuesta a Falla de Registro de Terminales Móviles
Fuente: [Elaboración Propia]



CAPITULO IV

Experimentación y Análisis de Resultados

14. CAPITULO IV EXPERIMENTACIÓN Y ANÁLISIS DE RESULTADOS

Para determinar si la hipótesis es aceptada o no lo llevamos a la siguiente prueba

14.1. EXPERIMENTACIÓN Y ANÁLISIS DE RESULTADOS

Luego de crear la aplicación Android, es necesario pasar a la etapa de evaluación de la misma. Para esta evaluación necesitamos realizar experimentos que nos permitirá ver el impacto en las personas que usen esta aplicación

Realizamos un análisis del impacto que causara la aplicación en las personas que lo usen y no lo usen.

Los grupos elegidos para la etapa del experimento son: las personas que usaran la aplicación y las personas que no usaran la aplicación.

14.1.1. DETERMINACIÓN DE LA MUESTRA

La población de estudio lo constituye en general todas aquellas personas que cuenten y no cuenten con la aplicación Android instalada en su terminal Smartphone Android. Esta población pertenece al Gobierno Autónomo municipal de la ciudad de La Paz.

La población característica de la muestra pertenece a personas que les interesa conocer la ubicación de una determinada persona, porque le interesa saber si se entra en un determinado perímetro o ha abandonado el mismo.

A continuación se muestran ejemplos en los que se requiere el control de ubicación de una persona.

Aplicaciones geriátrica: Cuando una persona envejece empieza a experimentar problemas de memoria, que se ven aún más acentuados con la aparición de la enfermedad de Alzheimer. Se quiere evitar los casos de personas que se han perdido porque no han sabido volver a casa, informando al servidor en todo momento de su situación.

Entorno empresarial: Muchas empresas utilizan móviles de empresa entre sus trabajadores. Por otro lado, es muy común llevar un control de entrada y salida de sus empleados para conocer las horas que han estado en el centro de trabajo. Si unimos ambos conceptos, se puede plantear la posibilidad de controlar a los trabajadores a través del teléfono móvil, y además facilitar el acceso a otros edificios del complejo empresarial, al abrirse las puertas al acercarse a una determinada distancia.

Niños: Se puede mantener a los padres informados de la situación de su hijo en una excursión o si se encuentran en el área de la escuela.

Cárceles: Teniendo en cuenta que Android se va a incluir en otros dispositivos, no necesariamente teléfonos móviles, existe la posibilidad de que se creen artilugios como un grillete inteligente que controle la ubicación de los presos en todo momento, alertando a las autoridades si hay peligro de fuga. Esta misma aplicación puede ser tenida en cuenta para los presos en tercer grado, o para aquellos casos en los que hay una sentencia de alejamiento.

Como se puede ver las situaciones en las que se pueden necesitar tener el control de un individuo son varias, además de las nuevas necesidades que vayan surgiendo.

14.1.2. PROCESO DE PRUEBA CON LA APLICACIÓN MÓVIL

Es necesario observar el impacto que causara la utilización de la aplicación móvil.

Resultados de evaluación ordinaria	Evaluación	Observación
71 a 100	Excelente	Muy Bueno
51 a 70	Bueno	Bueno
21 a 50	Regular	Satisfactorio
0 a 20	Malo	Deficiente

Tabla 4.1: Escala de Evaluación de la Aplicación Móvil
Fuente: [Elaboración Propia]

14.1.3. EVALUACIÓN DE RESULTADOS

Se forma dos grupos, uno experimental y el otro de control con la finalidad de someter ambos grupos en una misma prueba, de donde obtenemos los siguientes resultados:

Llamaremos entonces:

GRUPO V: Personas que Usan la Aplicación Móvil

GRUPO T: Personas que no Usan la Aplicación Móvil

Con la participación de las personas se obtiene la tabla 4.2

Persona	Grupo V (personas + aplicación)	Grupo T (personas sin aplicación)	Diferencia
1	70	35	35
2	80	60	20
3	70	40	30
4	85	65	20
5	75	60	15
6	80	30	50
7	60	43	17
8	95	77	18
9	80	64	16
10	100	90	10

Tabla 4.2: Resultados de la Evaluación de la Aplicación Móvil
Fuente: [Elaboración Propia]

Tabla 4.2: Comparación de Resultados de la Evaluación de la Aplicación Móvil
Fuente: [Elaboración Propia]

❖ **Personas que no utilizaron la aplicación móvil**

Luego el puntaje promedio obtenido por este grupo es:

$$X_T = \frac{\sum_{t=1}^n X_t}{n} = \frac{35 + 60 + 40 + 65 + 60 + 30 + 43 + 77 + 64 + 90}{10} = 56.40$$

Calculando la desviación estándar

$$S_t^2 = \frac{1}{n} \sum_{t=1}^n (x_i - x_T)^2 = 18.20$$

❖ **Personas que utilizaron la aplicación móvil**

Luego el puntaje promedio obtenido por este grupo es:

$$X_v = \frac{\sum_{t=1}^n X_t}{n} = \frac{70 + 80 + 70 + 85 + 75 + 80 + 60 + 95 + 80 + 100}{10} = 79.50$$

Calculando la desviación estándar

$$S_v^2 = \frac{1}{n} \sum_{t=1}^n (x_i - x_T)^2 = 11.28$$

14.1.4. PRUEBA DE HIPÓTESIS

Recordando la hipótesis:

“La implementación de una aplicación con interface Google Maps para la localización y monitoreo de dispositivos SmartPhone Android optimiza el control de personas”.

Para lo cual consideramos las hipótesis

H_0 : El grupo de las personas que no usen la aplicación Android lograrán mejores resultados para realizar la localización y monitoreo de personas.

H_1 : El grupo de personas que usen la aplicación Android logran mejores resultados para realizar la localización y monitoreo de personas.

Ahora procedemos a calcular el valor de t mediante la siguiente formular:

$$t = \frac{X_V - X_T}{\sqrt{\frac{s_V^2}{n_V} + \frac{s_T^2}{n_T}}}$$

Donde x_v es la media del grupo de persona que utiliza la aplicación móvil s_v^2 es la desviación estándar elevada al cuadrado y n_v es su tamaño, x_t es la media del grupo de personas que no utilizan la aplicación móvil s_t^2 es la desviación estándar elevado al cuadrado y n_t es su tamaño. Y el denominador es el error estándar de la distribución muestral de la diferencia entre medias.

Grupos	N	Media	Desviación Estándar
Grupo V	10	79.50	11.28
Grupo T	10	56.40	18.20

Tabla 4.3: Medias de tendencia central acerca de la interactividad

Fuente: [Elaboración Propia]

Calculando:

$$t = \frac{79.50 - 56.40}{\sqrt{\frac{(11.28)^2}{10} + \frac{(14.88)^2}{10}}}$$

$$t = 3.421$$

De donde:

t	Diferencia de Medias	Error estandar de la distribucion muestral
3.40	23.10	6.77

Tabla 4.4: Resultados del cálculo de valor t

Fuente: [Elaboración Propia]

Para saber si el valor es significativo, se aplica la formula y se calculan los grados de libertad. La prueba t se basa en la distribucion muestral o poblacional de diferentes medias, conocida como la distribucion t de student.

Esta Distribucion se identifica por los grados de libertad, los cuales contituyen el numero de maneras en que los datos puedan variar libremente. Son determinantes, ya que nos indican que valor debeos esperar de t, dependiendo del tamaño de los grupos

que se comaran. Cuanto mayor el número de grados de libertad se tengan, la distribución t de student se acerca mas a ser una distribución normal.

Ahora se procede al calculo de los grados de libertad, mediante la siguiente formula.

$$GL = (n_v + n_T) - 2$$

$$GL = (10 + 10) - 2$$

$$GL = 18$$

Ahora asumimos un nivel de confianza de o 5% de posibilidad de error. Cuando mayor sea el valor t calculado respecto al valor de la tabla y menor sea la posibilidad de error, mayor sera la certeza de los resultados.

Al hacer referencia a la tabla de distribución t student, buscamos los grados de libertad correspondientes y encontramos los siguientes valores:

Grados de Libertad (GL)	Nivel de Confianza		
	0.05	0.025	0.01
18	1.7341	2.1009	2.524

**Tabla 4.5: Distribución t de student para 18 grados de libertad
Fuente: [Elaboración Propia]**

Nuestro valor calculado de t es 3.4021 y resulta superior al valor t de tablas entonces aceptamos la hipótesis H_1 de la investigación y rechazamos la hipótesis H_0 , con nivel de confianza de 0.05 ($3.4021 > 1.7341$) el cual tiene un porcentaje del 95 % de aceptación y 5% de rechazo.



CAPITULO V

Conclusiones y Recomendaciones

15. CONCLUSIONES

En este trabajo se ha creado una aplicación de localización y control de terminales móviles Android. Esta aplicación nos permite conocer la ubicación de una terminal móvil indiferentemente de la persona que lo tenga, individuo que previamente ha sido informado de ello. Se ha buscado crear una aplicación sencilla pero que a la vez cumpla todas las funciones especificadas. El desarrollo completo de la aplicación y la página web se ha llevado a cabo con herramientas de licencia gratuita.

Los mayores problemas que han surgido a lo largo de la realización del trabajo Vienen derivados de la falta de experiencia en la plataforma Android, el diseño de la interfaz mediante el lenguaje XML fue muy dificultoso en un inicio, pero finalmente se acabó dominando éste aspecto. Otro de los puntos de mayor dificultad fue codificar la conexión con la base de datos desde la terminal móvil, ya que ha sido la primera experiencia en este trabajo. Fue de gran ayuda la gran variedad de tutoriales existentes en la red, tanto en español como en inglés.

Observamos que la hipótesis y los objetivos planteados en el presente trabajo de investigación se cumplen muy satisfactoriamente.

Cualquier idea que surja y que cumpla el objetivo principal de este trabajo, tiene como objetivo mejorar la aplicación y el de aprender cada día más.

16. RECOMENDACIONES

Se recomienda realizar la aplicación para otras plataformas móviles populares como iOS, Windows Phone, Symbian, BlacBerry.

Al haberse realizado el proyecto en el entorno de desarrollo de Eclipse el más utilizado en la actualidad, Se recomienda explorar Android Studio un nuevo entorno de desarrollo integrado para el sistema operativo Android lanzado por Google y puede ser descargada gratuitamente desde [<http://developer.android.com/sdk/installing/studio.html>].





Bibliografía

~~Roger S. Pressman, (2006). Ingeniería de Software un enfoque práctico. Editorial McGraw-Hill, 6ta Edición. ISBN: 970-10-5473-3~~

Jorge Santiago Nolasco Valenzuela, (2013): Desarrollo de Aplicaciones Móviles con Android. Editorial Macro EIRL, 1ra Edición. ISBN: 978-612-304-106-9

Jeff “JavaJeff” Friesen, (2010): Java para desarrollo Android. Editorial Anaya Multimedia, 1ra Edición. ISBN: 978-84-415-2961-8

Ronal Walpole y Raymond Myers, (2001), Probabilidad y Estadística. Editorial McGraw-Hill

Hernandez, R (2005): Metodología de la Investigación. Editorial McGraw-Hill Interamericana, 3ra Edición. ISBN: 9788484270775

Antonio Amescua, (2003): Análisis y Diseño estructurado y Orientado a Objetos de Sistemas Informáticos. McGraw-Hill Interamericana de España. ISBN: 9788448139247

Ian Sommerville, (2005): Ingeniería de Software. Editorial Addison-Wesley. ISBN 9788478290741

Referencias WEB:

Comunidad Latinoamericana de Metodologías Agiles, 2006. “metodologías agiles”. [Disponible en:

[http://es.wikipedia.org/wiki/Desarrollo_C3A1gil de software](http://es.wikipedia.org/wiki/Desarrollo_C3A1gil_de_software)] [25/11/2013]

Agiles”. [Disponible en : www.agiles.org] [25/11/2011].



Anexos
