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

TESIS DE GRADO

"SOFTWARE COMO SERVICIO PARA LA PREVENCIÓN DE CRIMINALIDAD URBANA UTILIZANDO MÉTODOS DE GEORREFERENCIACIÓN"

PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMATICA
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: JOSE LUIS CAMPERO PEREZ
TUTOR METODOLOGICO: M. Sc. ALDO RAMIRO VALDEZ ALVARADO
ASESOR: LIC. JAVIER REYES PACHECO

LA PAZ – BOLIVIA
2015

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

Dedicado a mis papás Rogelio y Miriam Julia, a mis hermanos Lorena, Carolina y Joe, que están a mi lado en las buenas y malas.

Mami en especial a ti que siempre tienes fe en mí, y no dejas de creer y darme ánimos para seguir y seguir en la vida. Porque me enseñas que hay solución a todo, y que la bondad aún existe. Gracias por estar aquí.

AGRADECIMIENTOS

A: Dios por darme la oportunidad de estar acá de vivir, de creer, de seguir luchando.

A: La Universidad Mayor de San Andrés, Facultad de Ciencias Puras y Naturales, carrera de Informática, que ha sido un segundo hogar para mí, y me permitió lograr mis sueños universitarios, y desarrollar mi criterio en busca de la verdad.

A: Mi tutor M. Sc. Aldo Ramiro Valdez Alvarado, por su comprensión, dedicación con su trabajo, expresión sabia para subir la moral de los estudiantes, guía de desarrollo y conclusión de la presente tesis.

A: Mi revisor Lic. Javier Hugo Gustavo Reyes Pacheco, por su criterio de la vida ante las situaciones buenas y malas que nos trae, por sus consejos, apoyo total y confianza a mi persona en el desarrollo de la presente tesis.

A: Los docentes que me dieron su enseñanza y pude sacar todo lo positivo en ellos.

A: Mis amigos, si a ustedes que siempre están ahí, quisiera nombrarlos a todos, pero saben que son ustedes y que siempre hay algo bueno que recordar.

A: Mis hermanos, Roberto, Carlos, Henry y Judith a pesar del tiempo seguimos unidos como verdaderos hermanos.

A: La persona que lee esto, te digo, "No te Rindas, por que donde hay fe hay posibilidad de realizar lo imposible" de todo Corazón te deseo éxitos en lo que te propongas.

A: La Música, al gran equipo de Futbol "The Strongest", Videojuegos, DC comics, cosas que son parte de mí, y me dieron todo tipo de experiencias, valores y fuerzas.

"Gracias Totales"...

RESUMEN

Con el paso del tiempo el interés por desarrollar aplicaciones móviles con servicios en la nube ha ido incrementando exponencialmente, y es un hecho que muchas personas cuentan con un Smartphone y el uso del internet es constante en ellos.

Es por eso que la tesis presente tiene la finalidad de realizar un software como servicio dentro de una aplicación móvil, cuyo funcionamiento cuenta con el apoyo de un servidor en la nube, en cual ofrece un servicio de información para los usuarios enfocado en la prevención de incidentes y delincuencia en la ciudad, realizando como prioridad el control de lugares de alto índice de criminalidad con ayuda de georreferenciación en un mapa, ubicación del usuario a través de la geolocalización, reporte de incidentes en determinadas horas del día en distintos lugares, Reporte de denuncias por el usuario.

El uso de información y las nuevas tecnologías han dado como resultado nuevas investigaciones en cuanto a métodos modelos y tecnologías, incluso en las herramientas para su mejor uso y facilidades de desarrollo e implementación. Es por eso que la metodología usada para el desarrollo de dicho software es Mobile – D que está enfocado en el desarrollo de aplicaciones móviles, respectivamente se utiliza WEBML para el modelado de esta aplicación y su descripción.

En cuanto a la demanda del Software se realizó una encuesta enfocando la necesidad de utilizar una aplicación móvil para la prevención e información de incidentes y crímenes en la ciudad.

La presente tesis se propone aminorar los índices de criminalidad urbana en determinado lugar con métodos de georreferenciación, con el uso de servicios móviles en la nube a través de una aplicación en su celular.

ABSTRACT

Over time the interest in developing mobile applications with cloud services has been increasing exponentially, and it is a fact that many people have a Smartphone and Internet use is constant in them.

That is why the present thesis aims to make software as a service in a mobile application, the operation has the support of a server in the cloud, which provides an information service for users, focused on prevention incidents and crime in the city, making it a priority to control places high crime rate using georeferencing on a map, user location through geolocation, incident reporting at certain times of day in different places, Report user complaints.

The use of information and new technologies have led to new research methods in terms of models and technologies including tools for better use and development facilities and implementation. That's why the methodology used for the development of such software is Mobile - D is focused on the development of mobile applications, respectively WebML used for modeling this application and its description.

As for demand Software, a survey focusing on the need for a mobile application for the prevention and reporting of incidents and crimes in the city was conducted.

This thesis aims to reduce urban crime rates in certain place with georeferencing methods, with the use of mobile cloud services through an application on your smart phone.

ÍNDICE

CAPÍTULO I
MARCO INTRODUCTORIO

1.1 Introducción 1
1.2. Antecedentes 2
1.2.1. Antecedentes de Cloud Computing 2
1.2.2. Georreferenciación y Geolocalización 5
1.2.3. Tesis Similares 6
1.3. Planteamiento del Problema 7
1.3.1. Problema Central 7
1.3.2 Problemas Secundarios 7
1.4. Definición de Objetivos 8
1.4.1. Objetivo General 8
1.4.2. Objetivos Específicos 8
1.5. Hipótesis 9
1.6. Justificación 9
1.6.1. Justificación Económica 9
1.6.2. Justificación Social 9
1.6.3. Justificación Científica 10
1.7. Alcances y Límites 10
1.7.1. Alcances 10
1.7.2 Límites 11
1.8. Aportes 12
1.8.1. Aporte Teórico 12
1.8.2. Aporte Práctico 13
1.9. Metodología 13
1.9.1. Metodología de Investigación 13
1.9.2. Metodología de Ingeniería 14

CAPÍTULO II
MARCO TEÓRICO

2.1. Ingeniería de Software 15
2.1.1. Metodologías Ágiles 15
2.1.2. Metodología de Programación Extrema XP 15
2.1.3. Modelo Vista Controlador (Model view controler) - MVC 16
2.1.4. Modelo Vista – Vista Modelo (Model view view Model) - MVVM 17
2.1.4.1. Los componentes del patrón MVVM 18
2.2. Ingeniería móvil 19
2.3. Metodología Mobile – D 19
2.3.1. Fase de exploración 20

2.3.2. Inicialización 24
2.3.3. Producción 27
2.3.4. Estabilización 30
2.3.5. Pruebas 33
2.4. Ingeniería Web 37
2.4.1. Modelado de Lenguaje Web - WEBML 37
2.5. Cloud computing 50
2.6. SAAS (Software como Servicio) 51
2.7. Windows Azure 52
2.8. Georreferenciación 54
2.8.1. Sistemas de Coordenadas y Datum Geodésicos 54
2.8.2. Métodos de Georreferenciación 54
2.9. Prevención de Criminalidad Urbana 55

CAPÍTULO III
MARCO APLICATIVO

3.1. Introducción 56
3.2. Fase de exploración 57
3.2.1. Establecimiento del proyecto 57
3.2.2 Establecimiento de Grupos de Interes Stakeholders 58
3.2.2. Definición de alcance 58
3.3. Fase de Inicialización 59
3.3.1. Requerimientos Iniciales 59
3.3.2. Análisis de requerimientos iniciales 60
3.3.3. Módulos de la Aplicación 60
3.3.4. Planificación inicial 61
3.3.5. Planificación de Fases 65
3.3.6. Plan de Iteraciones 67
3.4. Fase de Producción y Estabilización 68
3.4.1. Diseño del software como servicio 68
3.4.2. Diseño del sistema con Webml 70
3.4.3. Desarrollo del software como servicio 76
3.5. Fase de Pruebas del Sistema 92
3.5.1. Prueba de Dispositivos. 93
3.5.2. Pruebas de Uso 93

CAPÍTULO VI
ANALISIS DE DATOS Y RESULTADOS

4.1. Prueba de Hipótesis 94
4.1.1. Contraste de Rachas de Wald - Wolfowitz 94
4.1.2. Desarrollo de la Prueba de Hipotesis 96

CAPÍTULO V
CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones..... 100
5.2. Recomendaciones 101

BIBLIOGRAFÍA

6. Bibliografía..... 102
ANEXOS

Anexo A – Árbol de Problemas
Anexo B – Árbol de objetivos
Anexo C – Matriz de Marco Lógico
Anexo D cronograma de avance
Anexo E
Encuesta propuesta y sus resultantes

ÍNDICE DE FIGURAS

Figura 2.1. Model View Controller 17
Figura 2.2. Model View View Model 18
Figura 2.3 Ciclo de desarrollo de Mobile – D 20
Figura 2.4 Composición de la fase de Exploración 22
Figura 2.5 Composición de la fase de Inicialización 25
Figura 2.6 Composición de la fase de Producción 29
Figura 2.7. Composición de la fase de Estabilización 31
Figura 2.8 Composición de la fase de Estabilización 32
Figura 2.9 Composición de la fase de Prueba 35
Figura 2.10 Modelado Estructural 40
Figura 2.11 Modelado de Hipertexto 41
Figura 2.12 Modelado de Hipertexto 50
Figura 2.13. Cloud Computing 51
Figura 2.14. Modelos de Servicio 52
Figura 2.15 Arquitectura de la infraestructura de Windows Azure 53
Figura 3.1: Planificación de la Implementación del Software como servicio FEELSAFE 56
Figura 3.2: Diagrama de Historia de Usuario 68
Figura 3.3: Diseño de software como servicio y Mobile Service 69
Figura 3.4: Modelo de Datos estructural FEELSAFE 71
Figura 3.5: Modelo de Hipertexto 72
Figura 3.6: Modelo de Hipertexto Modelado de la Base de datos 78
Figura 3.7: Modelo de Hipertexto Conexión con el Servidor 80
Figura 3.8: Modelo de Hipertexto Interfaz de Usuario 82
Figura 3.9: Modelo de Hipertexto vista de mapa y sus características 85
Figura 3.10: Modelo de Hipertexto Historial delincuencial 89
Figura 3.11: Modelo de Hipertexto Reporte de zonas y calles evaluadas. 92

ÍNDICE DE TABLAS

Tabla 2.1. Modelos de diseño de hipertexto de WEBML 49
Tabla 3.1.: Cronograma de Avance del Proyecto 58
Tabla 3.2.: Requerimientos Iniciales Identificados 60
Tabla 3.3.: Historia de Usuario Uno 62
Tabla 3.4.: Historia de Usuario dos 62
Tabla 3.5.: Historia de Usuario tres 63
Tabla 3.6.: Historia de Usuario cuatro 63
Tabla 3.7.: Historia de Usuario cinco 64
Tabla 3.8.: Historia de Usuario seis 64
Tabla 3.9.: Historia de Usuario siete 65
Tabla 3.10.: Requerimientos Iniciales Identificados 67
Tabla 3.11.: Pantalla de Bienvenida 73
Tabla 3.12.: Acceso a la Aplicación 73
Tabla 3.13.: Pantalla de Datos georreferenciales 74
Tabla 3.14.: Pantalla Segmentación del mapa por zona 74
Tabla 3.15.: Pantalla de Denuncia de Usuario 75
Tabla 3.16.: Pantalla de listado de Usuarios 75
Tabla 3.17.: Pantalla de Lista de Lugares 76
Tabla 3.18.: Historia Diseño de la Base de Datos 77
Tabla 3.19.: Tarjeta de Tarea Diseño de la Base de Datos 77
Tabla 3.20.: Hoja de Prueba de Diseño de la base de datos 78
Tabla 3.21.: Historia Conexión con el Servidor 79
Tabla 3.22.: Tarjeta de Tarea Conexión con el servidor 80
Tabla 3.23.: Hoja de Prueba de Diseño de la base de datos 80
Tabla 3.24.: Historia Diseño de la Interfaz de Usuario 81
Tabla 3.25.: Tarjeta de Tarea Diseño de la Interfaz de Usuario 82
Tabla 3.26.: Hoja de Prueba de Diseño de la Interfaz de Usuario 82
Tabla 3.27.: Historia Vista de Mapa y sus características 83
Tabla 3.28.: Tarjeta de Tarea Vista de Mapa y sus características 84
Tabla 3.29: Hoja de Prueba vista de mapa y sus características. 84
Tabla 3.30.: Historia referencia de probabilidad delincuencial de acuerdo a la hora 86
Tabla 3.31.: Tarjeta de Tarea referencia de probabilidad delincuencial hora 86
Tabla 3.32: Hoja de Prueba de referencia de probabilidad delincuencial hora 87
Tabla 3.33.: Historia Historial delincuencial 87
Tabla 3.34.: Tarjeta de Tarea Historial delincuencial 88
Tabla 3.35: Hoja de Prueba de Historial delincuencial 89
Tabla 3.36.: Historia Reporte de zonas y calles evaluadas 90
Tabla 3.37.: Tarjeta de Tarea Reporte de zonas y calles evaluadas 91
Tabla 3.38: Hoja de Prueba de Reporte de zonas y calles evaluadas 91
Tabla 4.1: Esperanza y Varianza 95
Tabla 4.2. Propuesta de software como servicio y rango de aceptación de usuario 97

CAPÍTULO I

MARCO INTRODUCTORIO

1.1 Introducción

Hoy en día las personas utilizan teléfonos inteligentes “Smartphone” para un uso diario, lo que nos permite entender que tener un dispositivo móvil ya no es un lujo sino una necesidad, viendo este punto de vista ya claro es necesario saber qué es lo que se puede hacer y ofrecer con el uso de dichos dispositivos que con el paso del tiempo van incrementando sus cualidades y también su demanda en el mercado.

Teniendo en cuenta el uso de aplicaciones para celulares, se han desarrollado múltiples metodologías y herramientas de las cuales hablaremos sobre “Cloud Computing” y sobre Software como servicio “Software as a Service (SaaS)”. Los cuales tienen una fuerte participación en el mercado debido a su mayor escalabilidad, alta disponibilidad, y reducción de costes operativos.

Ya aclarado el punto anterior, vemos la posibilidad de implementar soluciones a problemas cotidianos que se tienen en la sociedad, un claro ejemplo es la delincuencia en todas sus medidas, y observamos que un ataque hacia una persona puede ser prevenido con ayuda de sus dispositivos móviles.

Una solución es el uso de estadísticas y control de patrones delictivos, de los cuales se pueden sacar resultados de probabilidad de ser atacado en cierto lugar y a determinada hora del día.

Esto no sería posible sin una ubicación y el uso de mapas así que se plantea como solución la georreferenciación de la cual podemos encontrar muchas ventajas al momento de saber dónde nos encontramos y poder acceder a información sobre lugares peligrosos que atenten contra las personas por actos delincuenciales.

Encontrando esta relación entre programación en la nube “Cloud Computing”, “Smartphone”, “Georreferenciación” y el punto a tratar que es la “Prevención de Crímenes”, nace la idea de crear un software como servicio a través de Mobile Services que verifique las estadísticas de lugares que han sido marcados como lugares de delincuencia alta, media o baja.

La tesis propuesta es reducir la criminalidad y proporcionar información verídica que favorezca al usuario para prevenir un atentado contra su persona. Demostrando que se pueden dar soluciones óptimas a problemas cotidianos con el uso de la tecnología móvil, y otras ya mencionadas.

1.2. Antecedentes

1.2.1. Antecedentes de Cloud Computing

A continuación se muestra lo más notable dentro de la evolución de la idea Cloud Computing.

1960 - John McCarthy opinó que "algún día computación podrá organizarse como un servicio público.

1.2.1.1. Sistemas Distribuidos

El desarrollo de los sistemas distribuidos vino de la mano de las redes locales de alta velocidad a principios de 1970. Más recientemente, la disponibilidad de computadoras personales de altas prestaciones, estaciones de trabajo y ordenadores servidores ha resultado en un mayor desplazamiento hacia los sistemas distribuidos en detrimento de los ordenadores centralizados multiusuario.

1980 - Empezó la masificación, la descentralización y la personalización de las pc´s.

1.2.1.2. Clúster Computing

Hay dos conceptos muy manejados al tratar con tecnologías de clustering, son los de escalabilidad vertical y escalabilidad horizontal. Hablamos de escalabilidad vertical cuando tratamos con un gran sistema. La escalabilidad horizontal consiste en emplear muchas maquinas pequeñas e interconectadas para realizar una tarea determinada. Con un coste proporcionalmente bajo por máquina, es una solución muy barata para algunos tipos de aplicaciones, como servicios Web, correo o súper-computación.

Un clúster es un conjunto de computadoras interconectadas con dispositivos de alta velocidad que actúan en conjunto usando el poder cómputo de varias CPU'S en combinación para resolver ciertos problemas dados.

1.2.1.3. Grid Computing

La Grid Computing es una nueva forma de computación distribuida que fue concebido a mediados del año 1990 pero es a partir del 2000 que se han llevado progresos considerables en la construcción de dicha infraestructura que permite utilizar de forma coordinada todo tipo de recursos (entre ellos cómputo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado.

Las ideas de grid (incluidas aquellas de la computación distribuida, programación orientada a objetos y servicios Web) fueron reunidas por Ian Foster, Carl Kesselman, y Steve Tuecke, conocidos como los "Padres de la Grid".

1990 - El termino nube fue robado por la telefonía con las redes privadas virtuales (VPN) con calidad de servicio pero a un menor precio, al equilibrar la utilización de la red su ancho de banda con más eficacia. Tiempo del modelo: (cliente–servidor), entre ellos Interactúan. Algunas funciones están en el cliente, otras en el servidor. La inteligencia es compartida. (Cliente - Servidor).

2000 - Similar al modelo cliente-servidor, pero otorgando mayor inteligencia al servidor. Luego fue el tiempo de Internet. La inteligencia seguía siendo compartida, cliente- internet.

1.2.1.4. Cloud Computing

Derivada del término cloud (Nube), usado como metáfora de infraestructuras tecnológicas complejas y cuyo origen se remite a la década de los 90, en referencia a las ya enormes redes ATM (Asíncronos Transfer Mode).

En el año de 1999, Marc Benioff, Parker Harris y otros socios, fundaron la compañía Salesforce.com, aplicando tecnologías desarrolladas por compañías como Google y Yahoo! a diversas aplicaciones de negocio. Ellos fortalecieron la entrega de servicios bajo demanda, particularmente SaaS, viéndose respaldados por miles de clientes y negocios exitosos.

A inicios del año 2000, Yahoo! y Google anunciaron la prestación de servicios cloud a cuatro de las más grandes universidades de Estados Unidos: la Universidad de Carnegie Mellón, la Universidad de Washington, la Universidad de Stanford y el Massachusetts Institute of Technology (MIT). Poco tiempo después IBM Corp. anunció el ofrecimiento de servicios cloud, seguido por gigantes informáticos como Microsoft, Oracle, Intel, SUN, SAS y Adobe, cuyos enfoques abarcaron la provisión de modelos IaaS, PaaS y SaaS.

2005 - Llega Cloud Computing. Gracias a nuevas tecnologías e infraestructura. La inteligencia y el servicio empezaron a estar en la nube de Internet. Gran cantidad de datos y su almacenamiento generan más y mejor inteligencia.

2006 - Amazon dio el gran paso cuando modernizo sus centros de datos, entonces las redes de computadoras solo utilizaban el 10% de su capacidad. Amazon Web Services popularizó el modelo IaaS, convirtiéndolo en una de las nociones principales de cloud computing. Su novedosa estrategia permitió la ejecución personalizada y bajo demanda de máquinas virtuales Linux en infraestructuras computacionales con una complejidad totalmente oculta a los usuarios finales

2007 - Google, IBM, y un número de universidades se embarcó en una escala de computación nube gran proyecto de investigación.

2008 - Eucalyptus se convirtió en el primero de código abierto compatible con AWS, API de la plataforma para el despliegue de nubes privadas. Gardner vio una oportunidad para la computación en nube "para dar forma a la relación entre los consumidores de servicios de TI, quienes utilizan los servicios de TI y aquellos que los venden".

2010 - Microsoft declaro que "Alrededor del 75 por ciento de nuestra gente está haciendo todo nube basado o inspirado por completo de nubes, dentro de un año que será de 90 por ciento. "

1.2.2. Georreferenciación y Geolocalización

El uso de la georreferenciación como herramienta analítica de la delincuencia se inició en 1942 cuando Shaw y McKay descubrieron que los delitos no se distribuyen al azar en el espacio urbano. El análisis de arrestos juveniles –ocurridos entre 1900 y 1933 en la ciudad de Chicago- mostró la existencia de áreas urbanas en las cuales estas tasas se mantenían estables, pese a que se observaba una alta rotación de sus residentes. Se concluyó entonces, que la localización de delitos en determinadas áreas urbanas se veía afectada por las características del contexto ecológico y estos lugares fueron denominados como “áreas desviadas” o “deviant places”.

1.2.2.1. Bing Maps

Anteriormente Live Search Maps, Windows Live Maps, Windows Live Local y MSN Virtual Earth) es un página de Mapeo Web creado por Microsoft para su buscador Bing. Su principal competencia es Google Earth.

Se basó en tecnologías existentes de Microsoft, como Microsoft MapPoint y TerraServer. La versión original carecía de muchos de sus características que podemos distinguir ahora;

incluyendo la Vista de Pájaro, mapas en 3d y la funcionalidad de deudas se limitaba a Bloc de un único "notas" de los puntos de interés. Su lanzamiento fue en diciembre de 2005, Windows Live Local en la cara pública de la plataforma de Virtual Earth. El 6 de noviembre de 2006, Microsoft añadido la capacidad de ver los mapas en 3D mediante un control de .NET administrado y gestionado a la interfaz Direct3D. Microsoft posteriormente oficialmente denomina este producto "Live Search maps", integrándolo como parte de su servicio de Live Search. El 3 de junio de 2009, Microsoft renombra oficialmente a Live Search Maps como Bing Maps y a la plataforma de Virtual Earth como Bing Maps for Enterprise.

1.2.3. Tesis Similares

Tesis de grado: "Georreferenciación y su uso en Smartphone para la localización de personas", **Institución UMSA**, **Autor: Jorge Marcelo Cupi Veliz.** Este proyecto da un servicio de geolocalización que requiera un celular que cuenta con el servicio de GPS para poder localizar personas. El proyecto utiliza una metodología ágil propuesta por el mismo autor denominada MAD (Mobile Agile Development) desarrollo ágil y móvil, y desarrolla la aplicación bajo la metodología propuesta.

Tesis de grado: "Sistema de información gerencial para el control y seguimiento de las especies silvestres en Áreas Urbanas: Caso La Paz", Institución UMSA, Autor: Luis Alberto Condori Chambi. Lo que plantea este proyecto es Mantener la información y el seguimiento de las especies de la urbe paceña, a través de un sistema de información georreferencial utilizando Google Maps. El proyecto utiliza la metodología ágil XP.

Tesis de grado: "Software como servicio para la administración de historias clínicas", Institución UMSA, Autor: Yecid Alejandro Yahuita Ponce. El planteamiento de la tesis de desarrollar un Software como Servicio, para la administración de historias clínicas para que médicos tengan un mejor control y seguimiento de sus pacientes. La metodología usada en esta Tesis es OpenUp, interpretada como metodología ágil.

1.3. Planteamiento del Problema

1.3.1. Problema Central

De acuerdo a los datos proporcionados por la Dirección Nacional de la FELCC, entre 2011 y 2013 se ha registrado un total de 762 pandillas juveniles en las ciudades capitales de departamento (excepto Trinidad). En la ciudad de La Paz (zona central y zona sud) se encuentra el 34% de las pandillas; en Santa Cruz el 22% y en Cochabamba el 14%. En Oruro y El Alto, se encuentra algo más del 17% de estas agrupaciones. La población masculina representa el 79% y la femenina el 21%.

Lo primero que debemos de preguntarnos es que hacen las autoridades sobre este problema, y también que hacemos nosotros para resolverlo. Puesto a que no existe ningún método de prevención de asaltos a no ser las que por lógica conocemos, es decir no estar afuera en altas horas de la noche o buscar lugares más alumbrados y más concurridos para caminar por la noche.

Lo que conlleva a un índice bastante grande de criminalidad, no por el crecimiento de delincuencia, si no por falta de soluciones, que están al alcance de nuestras manos, lo que conlleva a la siguiente cuestión como problema central:

¿Cómo identificar los problemas de delincuencia urbana y prevenir incidentes de acuerdo al lugar donde suceda?

1.3.2 Problemas Secundarios

A continuación se muestran los problemas secundarios encontrados a partir del problema central:

- La escasa información sobre crímenes efectuados en determinado lugar, es lo que ha causado muchos problemas a la ciudadanía en cuanto a su seguridad.

- La no prevención de lugares de recorrido seguro, provoca el crecimiento de víctimas de abuso delincuencial.
- La falta de soluciones tecnológicas en cuanto a la seguridad peatonal, provoca el desinterés de la ciudadanía para optar a nuevas herramientas de precaución.
- El poco análisis en el uso de métodos de georreferenciación y la delincuencia, es lo que ocasiona bajos resultados en las investigaciones criminales.
- La falta de medios de difusión digital donde se pueda hacer denuncias acerca de un delito, ha perjudicado la información acerca de posibles zonas delincuenciales.

1.4. Definición de Objetivos

1.4.1. Objetivo General

Desarrollar un software como servicio para la identificación de criminalidad urbana y prevención de incidentes utilizando métodos de georreferenciación.

1.4.2. Objetivos Específicos

Como objetivos específicos tenemos los siguientes:

- Plantear un análisis estadístico sobre hechos delictivos ocurridos, que muestren información sobre crímenes efectuados en determinado lugar y reduzcan los problemas de inseguridad.
- Ofrecer una aplicación móvil a la ciudadanía como prevención delincuencial que determine un recorrido alterno para su seguridad y así reducir el índice de abuso delincuencial.
- Proponer una ayuda tecnológica a través de un servicio móvil cuanto a la seguridad peatonal que aliente el interés a los ciudadanos en usar herramientas tecnológicas de precaución para su seguridad.
- Plantear un método para el uso georreferenciación como análisis delincuencial, que posibilite el mejor manejo de la información criminal para los investigadores.

- Proporcionar un software en el cual se pueda destacar los delitos de manera segura y sin afectar la privacidad del denunciante, con esto prevenir a las personas sobre lugares peligrosos.

1.5. Hipótesis

El uso de Cloud Computing y Programación móvil, sobre una plataforma de servicio en la nube, facilita la información de mapas referido a la georreferenciación y geolocalización de manera remota e independiente, los cuales se utilizarán para dar soluciones a problemas de criminalidad y dar paso a la prevención de incidentes en la ciudad de La Paz.

1.6. Justificación

1.6.1. Justificación Económica

Los logros más destacados sobre la programación en la nube son las económicas, debido a que el ahorro en el mercado es muy conveniente para sus usuarios, no es necesaria la compra de equipos de última generación y está al alcance económico de la sociedad, sean dispositivos móviles de última generación o contemporánea los cuales gozan de equipamiento necesario para lo que queremos lograr que es la georreferenciación y geolocalización las cuales se pueden utilizar con ayuda de los dispositivos WI-FI o GPS. Integrados ya en los celulares inteligentes “Smartphone”.

1.6.2. Justificación Social

Vivimos en una época en la que la sociedad requiere de información al alcance de las manos y de hacer click en un botón de aplicación en su celular.

Dado este punto de vista el presente proyecto se encargara de brindar información acerca de zonas peligrosas a los usuarios de tal forma poder resguardar su seguridad ante posibles agresiones delincuenciales y traumáticas. Las personas que fueron víctimas de algún atentado delincuencial podrán contar con un software de denuncia y esto ayudara a otras

personas a prevenir posibles incidentes futuros, obteniendo los modus operandi de los criminales y los lugares donde se efectúa esto.

1.6.3. Justificación Científica

El uso de la programación en la nube "Cloud Computing" es un tema muy avanzado y utilizado en el entorno de la informática. Como también la programación de aplicaciones móviles que utilizan este servicio.

La georreferenciación para el análisis de crecimiento delincuencial no es debidamente utilizada por los programadores, si bien es un tema de la Policía e Investigación de la misma, la falta de explotación de este recurso es abismal dentro del ofrecimiento de aplicaciones ya sean web o móviles. Es por esto que se plantea como una solución holista la propia seguridad ciudadana de manera personal, a parte del apoyo policial ya antes mencionado.

1.7. Alcances y Límites

1.7.1. Alcances

Dentro de esta investigación se tomaran en cuenta distintos términos de estudio como ser el desarrollo de una aplicación móvil, la investigación estadística delincuencial y el planteamiento de un método de georreferenciación de análisis delincuencial, el tiempo estimado para el desarrollo de esta solución será aproximadamente de un año, dado es estudio y evaluación de la aplicación, realizado por los consumidores y a través de un "FeedBack" facilitado por la dirección de un correo electrónico proporcionado en el About de dicha aplicación.

El presente documento presenta los siguientes alcances espaciales del proyecto de acuerdo a las respectivas áreas en las que se efectuara:

Geolocalización:

- ✓ Ubicación del usuario en el mapa dentro la aplicación a través de geolocalización.
- ✓ Obtención del lugar en el que se encuentra el usuario en tiempo real.

Georreferenciación:

- ✓ Uso de técnicas de georreferenciación para el control delincuencial.
- ✓ Uso de técnicas de georreferenciación para la prevención criminal.

Estadístico:

- ✓ Obtención de índices delincuenciales en las zonas.
- ✓ Proporción de datos estadísticos según la hora.
- ✓ Proporción de informes acerca de los datos obtenidos acerca de la criminalidad en determinadas zonas.

Software como servicio:

- ✓ Descarga de la aplicación de carácter nacional, con posibilidad de expansión.
- ✓ Reporte de lugares de alto índice delincuencial.
- ✓ Registro del usuario a través de la aplicación.
- ✓ Registro de denuncias por parte de los usuarios frente a algún atentado delincuencial.
- ✓ Verificación actualizada de acuerdo al número de veces en las que el usuario se conecte a Internet.

1.7.2 Límites

A continuación se presentan lo que el software no hace o no considera hacer:

- ❖ Esta aplicación no solucionará los problemas de criminalidad, solo facilitara la información para una futura prevención.
- ❖ Esta aplicación no se hará responsable de denuncias falsas, en el registro de denuncias.

- El desarrollo de la aplicación no implica con la creación de cuentas en Facebook, debido a que se considera que el usuario ya tiene una.
- Esta aplicación no se hará responsable de la admisión de usuarios con falsa identidad.
- El software como tal no garantiza la seguridad del usuario solo le ayuda a prevenir futuros incidentes.
- La aplicación plantea un análisis delincuencial no proporcionado por un método o un estudio ya antes realizado.
- El software por ahora solo estará habilitado para la plataforma Windows Phone, y Windows 8.1 para adelante, a futuro multiplataforma.

1.8. Aportes

1.8.1. Aporte Teórico

La presente investigación tendrá como logro impulsar al estudio de problemas en la ciudadanía con ayuda de aplicaciones móviles y cloud computing, el cual no está muy explotado por ahora.

El software como servicio se verá empleado y desarrollado con la metodología ágil XP y Mobile – D ya que estas dos metodologías se relacionan muy bien entre si efectuando un rendimiento óptimo.

Para el modelado del lenguaje en el desarrollo de este proyecto se ha concluido utilizar WEBML el cual manifiesta de manera específica los módulos a utilizar en el desarrollo de "Software como Servicio".

Dentro del entorno de desarrollo Microsoft Visual Studio 2013 se utilizara el patrón de arquitectura MVVM (Modelo Vista Vista Modelo) el cual está basado en el modelo MVC (Modelo Vista controlador), el cual soporta el paradigma de la programación orientada a eventos.

Este servicio estará dedicado para aplicaciones móviles, utilizando Microsoft Azure para el almacenamiento no relacional, con la ayuda del entorno de desarrollo VS2013.

1.8.2. Aporte Práctico

El principal aporte de la presente investigación es desarrollar e implementar un software como servicio para la prevención de criminalidad urbana con ayuda de métodos estadísticos y georreferenciación que permita a los usuarios obtener información necesaria para contribuir a su propia seguridad.

Proporcionar información específica que detalle datos referenciales sobre índices delincuenciales en determinadas zonas y poder estudiarlas a partir de estos datos.

1.9. Metodología

1.9.1. Metodología de Investigación

1.9.1.1. Método Descriptivo

El método descriptivo utilizado por el presente proyecto es el método científico el cual es un proceso de etapas, que obtiene conocimiento valido: No es subjetivo, es verificable, es reducible, es reusable, es factible. Sus etapas cumplidas son: Observación la cual debe ser sistemática que tiene que ver con el problema, hipótesis la cual es una suposición que forma parte de una solución, experimentación, teoría y ley.

1.9.1.2. Estudio Descriptivo

El objetivo de este estudio consiste en llegar a conocer las situaciones, costumbres y actitudes predominantes a través de la descripción exacta de las actividades, objetos y personas. Su meta no se limita a la recolección de datos, sino a la predicción e identificación de las relaciones que existen entre dos o más variables. Con el fin de extraer generalizaciones significativas que contribuyan al conocimiento.

1.9.2. Metodología de Ingeniería

1.9.2.1. Método Mobile - D

Para dicha investigación dentro lo que es desarrollo de software se utilizara la metodología Mobile – D, la cual es un enfoque ágil para el desarrollo de aplicaciones móviles. Esta metodología tiene como objetivo, conseguir ciclos de desarrollo muy rápidos, el cual está basado en metodologías conocidas pero aplicadas de forma estricta como: Extreme Programing, Crystal Methodologies, Rational Unified Process.

1.9.2.2. Técnica

El uso del entorno de desarrollo colaborara en el diseño de interfaz de usuario, el cual permitirá el mejor manejo de este con la aplicación. Apoyado por el Modelo Vista Controlador (MVC) el cual es un patrón de diseño y arquitectura que básicamente separa en tres clases el modelado de datos. Y el patrón de arquitectura usado en la ingeniería de software desarrollado por Microsoft, que está fuertemente basado en el patrón MVC, orientado a plataformas de desarrollo basadas en interfaz de usuario, soportando el paradigma de la programación orientada a eventos “Model View View Model” (MVVM).

1.9.2.3. Herramientas

El software será desarrollado bajo las siguientes herramientas:

- Plataforma de desarrollo: Windows 8.1.
- Entorno de desarrollo: Microsoft Visual Studio 2013
- Plataforma de servicios de computación: Microsoft Azure
- Lenguaje de programación: C#.

CAPÍTULO II

MARCO TEÓRICO

2.1. Ingeniería de Software

La evolución de la disciplina de ingeniería de software ha traído consigo propuestas diferentes para mejorar los resultados de proceso de construcción, las metodologías tradicionales haciendo énfasis en la planeación, mientras las metodologías ágiles haciendo énfasis en la adaptabilidad del proceso delinean las principales propuestas presentes en la literatura. De manera paralela el tema de modelos para el mejoramiento de los procesos de desarrollo ocupa un lugar muy importante en la búsqueda de la metodología adecuada para producir software de calidad en cualquier contexto de desarrollo.

2.1.1. Metodologías Ágiles

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

En febrero del 2001, tras una reunión celebrada en Utah[1], nace el término "ágil" aplicado al desarrollo software. El objetivo fue esbozar los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y responder a los cambios que pueden surgir a lo largo del proyecto. Esto pretende ser una alternativa a los procesos de desarrollo tradicionales caracterizados por su total rigidez y muy dirigidos a la documentación que se genera tras cada una de las actividades desarrolladas.

2.1.2. Metodología de Programación Extrema XP

Las metodologías ágiles (como por ejemplo XP, SCRUM, DSDM, Crystal) forman parte del movimiento de desarrollo ágil de sotfware, que se basan en la adaptabilidad de cualquier cambio como medio para aumentar las posibilidades de éxito de un proyecto.

De forma que una metodología ágil es la que tiene como principios que:

- Los individuos y sus interacciones son más importantes que los procesos y las herramientas.
- El software que funciona es más importante que la documentación exhaustiva.
- La colaboración con el cliente en lugar de la negociación de contratos.
- La respuesta delante del cambio en lugar de seguir un plan cerrado.

2.1.3. Modelo Vista Controlador (Model view controler) - MVC

MVC o Modelo-Vista-Controlador por sus siglas en español es un patrón de diseño y arquitectura que separa los datos y la lógica de negocio de una aplicación de la interfaz de usuario y el módulo encargado de gestionar los eventos y las comunicaciones, que básicamente separa en tres clases el modelado de datos: Se tiene un dominio, una presentación y las acciones basándose en la entrada de datos del usuario:

- Modelo. El modelo gestiona los datos del dominio de aplicación y comportamiento, responde a las solicitudes de información sobre su estado (generalmente de la Vista) y responde a las instrucciones para cambiar el estado (generalmente desde el Controlador). Es la representación de la información con la cual el sistema opera, por lo tanto gestiona todos los accesos a dicha información, tanto consultas como actualizaciones, implementando también los privilegios de acceso que se hayan descrito en las especificaciones de la aplicación.
- Vista. Presenta el 'modelo' (información y lógica de negocio) en un formato adecuado para interactuar (usualmente la interfaz de usuario) por tanto requiere de dicho 'modelo' la información que debe representar como salida.
- Controlador. El controlador interpreta la entrada de datos del usuario, informando al Modelo o la Vista para realizar cambios según corresponda. Responde a eventos (usualmente acciones del usuario) e invoca peticiones al 'modelo' cuando se hace alguna solicitud sobre la información (por ejemplo, editar un documento o un registro

en una base de datos). También puede enviar comandos a su 'vista' asociada si se solicita un cambio en la forma en que se presenta el 'modelo' (por ejemplo, desplazamiento o scroll por un documento o por los diferentes registros de una base de datos), por tanto se podría decir que el 'controlador' hace de intermediario entre la 'vista' y el 'modelo'.

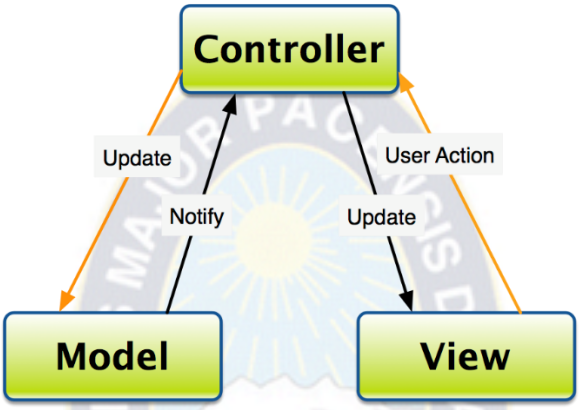


Figura 2.1. Model View Controller
Fuente: [Stackoverflow, 2011]

2.1.4. Modelo Vista – Vista Modelo (Model view view Model) - MVVM

Es un patrón de arquitectura usado en la ingeniería de software desarrollado por Microsoft, que está fuertemente basado en el patrón Modelo Vista Controlador (MVC), está orientado a plataformas de desarrollo basadas en la interfaz de usuario, que soportan el paradigma de la programación orientada a eventos.

MVVM es una variación de Martin Fowler 's patrón de diseño Presentación. Es como el modelo de presentación de Fowler, MVVM abstrae de una visión del estado y comportamiento. Sin embargo, mientras que abstrae la Presentación Modelo vistas (es decir, crea un vista de modelo) de una manera que no depende de una plataforma de interfaz de usuario específica, MVVM fue desarrollado por Microsoft arquitectos Ken Cooper y Ted Peters específicamente para simplificar la programación orientada a eventos de interfaces de

usuario explotar características de Windows Presentation Foundation (WPF) (de Microsoft .NET sistema de gráficos) y Silverlight (derivado de la aplicación de Internet de WPF).

2.1.4.1. Los componentes del patrón MVVM

Modelo.- Se refiere ya sea a un modelo de dominio, lo que representa el contenido verdadero estado (un enfoque orientado a objetos), o para la capa de acceso a datos que representa que el contenido (un enfoque centrado en los datos).

Vista, Vista modelo.- Al igual que en los patrones MVC y MVP, la vista es la interfaz de usuario (UI). El modelo de vista es una abstracción de la vista que expone las propiedades públicas y comandos. En lugar del controlador del patrón MVC, o el presentador del patrón MVP, MVVM tiene un ligante. En el modelo de vista, esta carpeta media la comunicación entre la vista y la carpeta de datos. El modelo de vista ha sido descrito como un estado de los datos en el modelo.

Carpeta.- De datos declarativos y comandos vinculantes están implícitos en el patrón MVVM. En la pila de soluciones de Microsoft, la carpeta es un lenguaje de marcado llamado XAML.

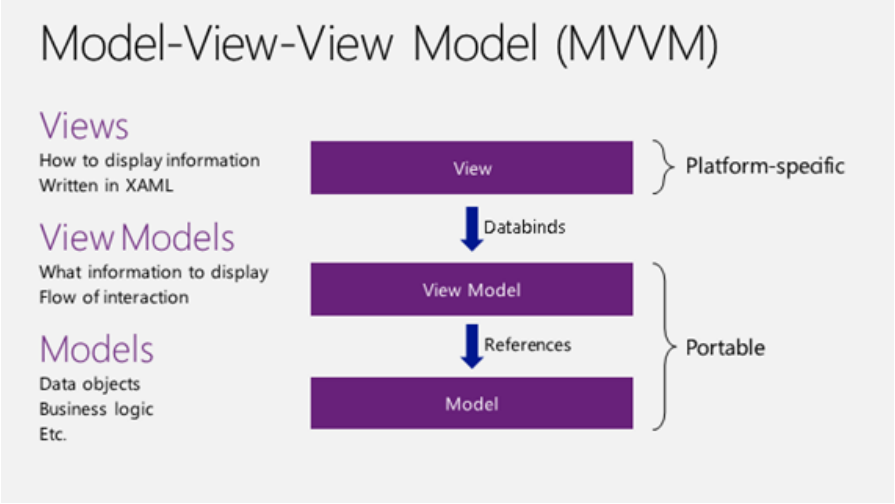


Figura 2.2. Model View View Model
Fuente: [Msdn, 2014]

2.2. Ingeniería móvil

Con el paso del tiempo la ingeniería de sistemas ha ido avanzando, hasta el punto de combinarse con la tecnología Smartphone, del cual se encuentran muchas facilidades para el usuario tanto como metodologías para esta rama.

La ingeniería móvil no es más que la combinación de la ingeniería en sistemas y su aplicación con la tecnología móvil, construyendo así una nueva forma de implementar sistemas de manera compacta y con ayuda de servicios web, servicios de aplicación, entre otros y con ayuda de la nube.

2.3. Metodología Mobile – D

Mobile D es un enfoque ágil para el desarrollo de aplicaciones móviles. Además de desarrollo de software es conveniente para varios contextos, por ejemplo, la seguridad, financiera, logística, y aplicaciones de productos de simulación.

El objetivo es conseguir tipos de desarrollos muy rápidos en equipos muy pequeños (de no más de diez desarrolladores) trabajando en un mismo espacio físico.

Según este método, trabajando de esa manera se deben conseguir productos funcionales en menos de diez semanas.

Está basada en Extreme Programming (Practicas), Metodologias de Cristal (escalabilidad) y Rational Unified Process (cobertura).

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. A continuación se muestra la metodología en la figura 2.1.

Figura 2.3 Ciclo de desarrollo de Mobile – D
Fuente: [Elaboración Propia]

2.3.1. Fase de exploración

En esta primera etapa, se define el proyecto la planificación y las partes involucradas, así como el equipo de trabajo y las fechas de entrega.

El propósito de la fase de Exploración es la planificación y el establecimiento del proyecto. "Un buen planeamiento a medio hacer" un dicho para ser recordado también en el contexto de desarrollo de software. La Fase de Exploración puede ser desapegado oportunamente en las últimas fases de Mobile - D y también se solapan con fases de 0 iteraciones.

Es una fase importante para establecer las bases para la implementación controlada de software en relación con el desarrollo de productos, por ejemplo, las cuestiones relacionadas con la arquitectura del producto, proceso de desarrollo de software y la selección de medio

ambiente. Diferentes grupos de StackeHolders son necesarios para ofrecer su experiencia en la fase de Exploración.

Los objetivos de la fase de Exploración son:

- Establecer los grupos de actores necesarios en la planificación y el seguimiento del proyecto de desarrollo de software.
- Definir y acordar los objetivos y el alcance del proyecto de desarrollo de software.
- Planificar el proyecto respecto del medio ambiente, el personal y cuestiones de procedimiento.

Los criterios de ingreso:

- Existe la propuesta de producto.
- Grupo de Exploración para el inicio del proyecto ha / han definido.
- El compromiso de la Administración para el proyecto se gana.

Entradas:

- La propuesta de producto.
- Biblioteca proceso de Mobile - D.
- Biblioteca de procesos de organización.
- Contrato.
- Requisitos iniciales documento.
- Plan de Proyecto.
- Las normas pertinentes (si los hay).

Criterios de salida:

- Grupo de clientes ha sido identificados.
- Los grupos de interés se han definido y establecido.
- Requisitos iniciales y el horario se han definido para el producto.

- El personal y el ambiente necesarios durante el desarrollo de productos se han seleccionado y asignado (por ejemplo, el personal de apoyo del equipo / formación / entrenadores, grupo de dirección / gestión, atención al cliente).
- La descripción del proceso se ha adaptado y documentado para el proyecto.
- Formación

Salidas:

- El documento de requisitos iniciales donde los requisitos iniciales han sido definido.
- Proyecto plan Incluyendo línea de tiempo, ritmo, terminaciones, recursos de proyecto / grupos de actores y sus responsabilidades.
- Base descripción del proceso que incluye el proceso de la línea de base, la garantía de calidad, actividades, documentación, puntos de integración.
- Plan de Medición.
- Plan de Formación.
- Arquitectura descripción de línea.

La siguiente figura ilustra las etapas de las cuales la fase Exploración se compone:

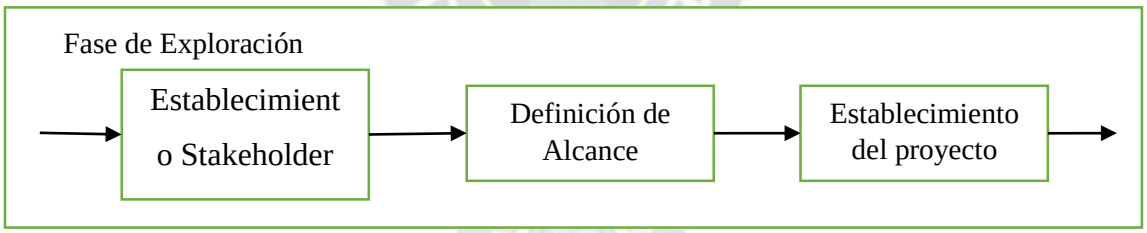


Figura 2.4 Composición de la fase de Exploración
Fuente: [Elaboración Propia]

2.3.1.1. Etapas

a) Establecimiento StakeHolder (Grupos de Interés).- Es una etapa en la que todo lo relevante de los StakeHolders - excluyendo el equipo del proyecto en sí - se necesita en el establecimiento, así como en diferentes tareas del proyecto son incipiente definida con las funciones y los recursos pertinentes. Además del cliente grupo (definido en el

Establecimiento Tarea patrón Cliente), el interesados en Mobile-D pueden incluir, por ejemplo, el grupo de dirección, gestión de proyectos, el grupo de arquitectura y especialistas de proceso. Todas estos actores juegan un papel vital en las tareas posteriores de Explora fase y en la ejecución del proyecto.

b) Definición de Alcance.- Es una etapa en la que los objetivos y el alcance del proyecto de desarrollo de software se definen y acordado por los StakeHolders. Esto incluye temas como la (inicial) requisitos para los productos y la línea de tiempo del proyecto.

c) Establecimiento del proyecto.- Es una etapa para acordar los asuntos ambientales del proyecto (físico y técnico), así como el personal necesario en desarrollo de software (desarrolladores y apoyo). Además, Las cuestiones del proceso se definen en esta etapa.

2.3.1.2. Patrones relacionados

La fase de Exploración se compone de los siguientes patrones de la etapa:

- **Establecimiento de StakeHolders (Grupos de Interes):** La identificación y asignación de los StakeHolders pertinentes son las primeras tareas de la fase de exploración.
- **Definición del Alcance:** Objetivos y alcance del proyecto deben ponerse de acuerdo sobre las primeras etapas del proyecto, incluyendo los requisitos y planificación de proyectos y problemas de arquitectura.
- **Establecimiento del Proyecto:** La etapa de establecimiento proyecto incluye la selección del medio ambiente para el proyecto, la asignación de personal y establecimiento de procesos.
- **Establecimiento de clientes:** El patrón de tareas Establecimiento Cliente es una instancia del patrón del establecimiento de StakeHolders. Debido al papel importante del cliente, ya sea dentro o fuera del establecimiento de clientes, la colaboración está incluido en Mobile

2.3.2. Inicialización

Configuramos el proyecto identificando y preparando todos los recursos necesarios. En esta fase la dedicaremos un día a la planificación y el resto al trabajo y publicación.

El propósito del modelo de fase de inicialización es permitir que el éxito de la próxima fases del proyecto mediante la preparación y verificación de todas las cuestiones fundamentales del desarrollo a fin de que todos están en plena disposición en el final de la fase de la aplicación de los requisitos seleccionados por el cliente.

Los objetivos del modelo de fase de inicialización son:

- Obtener una buena comprensión global del producto para el equipo de proyecto basado en los requisitos iniciales y descripciones de línea de arquitectura.
- Preparar los recursos físicos, técnicos y humanos, así como al cliente problemas de comunicación, planes de proyectos y todo desarrollo crítico para que todos ellos están en plena disposición para los requisitos de ejecución seleccionados por el cliente durante las próximas fases del proyecto.

Los criterios de ingreso:

- La fase precedente Explora se ha completado.

Entradas:

- Requisitos iniciales documento
- Plan de Proyecto.
- Descripción del proceso Base.
- Plan de Medición.
- Plan de Formación.
- Arquitectura descripción de línea,
- Cartera de productos.

Criterios de salida:

- El plan del proyecto se ha actualizado y un plan de línea de la arquitectura ha sido creado.
- Se han realizado los controles previstos.

Salidas:

- Actualización del plan de proyecto y un plan de línea de la arquitectura.
- La primera versión de Software Arquitectura y Diseño Descripción documento.
- Funcionalidad implementada.
- La primera versión de la cartera de productos.
- El documento de requisitos iniciales actualizada.
- Observa Desarrollador y la interfaz de usuario ilustraciones de cada requisito discutido.
- Las pruebas de aceptación para cada requisito.

El modelo de fase de inicialización puede llevarse a cabo a través de las siguientes etapas:

Figura 2.5 Composición de la fase de Inicialización
Fuente: [Elaboración Propia]

2.3.2.1. Etapas

a) Proyecto Set – Up.- El objetivo de esta fase es:

- puesta en marcha de los recursos físicos y técnicos para la proyecto, así como el medio ambiente para el proyecto monitoreo.

- Entrenar al equipo del proyecto, según sea necesario.
- Establecer las formas específicas de proyectos para comunicarse con el grupo de clientes.

b) Planificación Inicial.- El objetivo esta fase es:

- Obtener una buena comprensión global del producto para el equipo del proyecto sobre la base de los requisitos iniciales y descripciones de línea arquitectura.
- Perfeccionar y el detalle de la línea arquitectura descripciones y el plan del proyecto.
- Crear plan de línea de la arquitectura.
- Preparar planes para comprobar el estado de preparación de cuestiones fundamentales del desarrollo, tales como el desarrollo medio ambiente, elementos arquitectónicos importantes y otras entidades externas del software a desarrollar así como la comunicación entre los elementos y entidades.

c) Día de Trabajo en Iteración cero (0).- El propósito de esta etapa son a las verificaciones previstas de desarrollo crítico problemas y para resolver las cuestiones pendientes, si es posible.

2.3.2.3. Patrones relacionados

Otros patrones que forman “parte de”, “compuestos por”, o “asociados” estrechamente con la fase de inicialización son.

a) **El patrón de fase Exploración** se utiliza para implementar la fase de exploración que proporciona insumos para esta fase.

b) Este modelo utiliza el Proyecto SetUp, Planificación inicial, y el Día de Trabajo en cero (0) patrones de iteración para llevar a cabo el patrón.

c) **El patrón Producción** se utiliza para implementar la fase Producción que es el sucesor de la fase de inicialización.

2.3.3. Producción

El propósito en la fase Producción es implementar la funcionalidad requerida en el producto mediante la aplicación del ciclo de desarrollo iterativo e incremental.

Se repiten iterativamente las sub fases. Se usa el desarrollo dirigido por pruebas, 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.

Los objetivos del modelo de fase de producción son:

- Implementar el cliente priorizado funcionalidad al producto.
- Centrarse en la funcionalidad básica fundamental por su aplicación a principios del incrementar para permitir múltiples ciclos de mejora.

Los criterios de ingreso:

- La fase anterior a 0 iteración se ha completado.
- Se han identificado los requisitos funcionales más importantes.
- El equipo se ha reunido y ha sido entrenado para el método de desarrollo.
- El entorno de desarrollo ha sido establecido.

Entradas:

- Actualización del plan de proyecto y un plan de línea de la arquitectura.
- La primera versión de Software Arquitectura y Diseño Descripción documento.
- Los planes para la comprobación de las cuestiones fundamentales del desarrollo.
- Funcionalidad implementada.
- Cartera de productos que contiene todos los requisitos identificados del proyecto (tanto funcional como no funcional).
- Métricas de datos.
- La experiencia del equipo del proyecto.

- Documentación de prueba de aceptación.
- Historia y tarjetas de tarea.
- Los datos sobre los recursos gastados.
- Manuales, especificaciones API y otro material de apoyo.
- Pruebas unitarias.

Criterios de salida:

- La funcionalidad más importante ha sido implementado

Salidas:

- Funcionalidad implementada.
- Documentación de pruebas de aceptación.
- Notas de Desarrollo.
- IU ilustraciones.
- Lista de puntos de acción.
- Actualización cartera de productos.
- Actualizado plan del proyecto.
- Historia y Tarjetas de tarea.
- Los datos Experiencia del equipo de proyecto.
- Conocimiento de los requisitos del sistema y pruebas de aceptación.
- Lista de defectos.
- Lanzamiento lista de verificación de auditoría.
- Requisitos iniciales documento.
- Informe de estado diario

El modelo de fase de producción puede llevarse a cabo a través de las siguientes etapas:

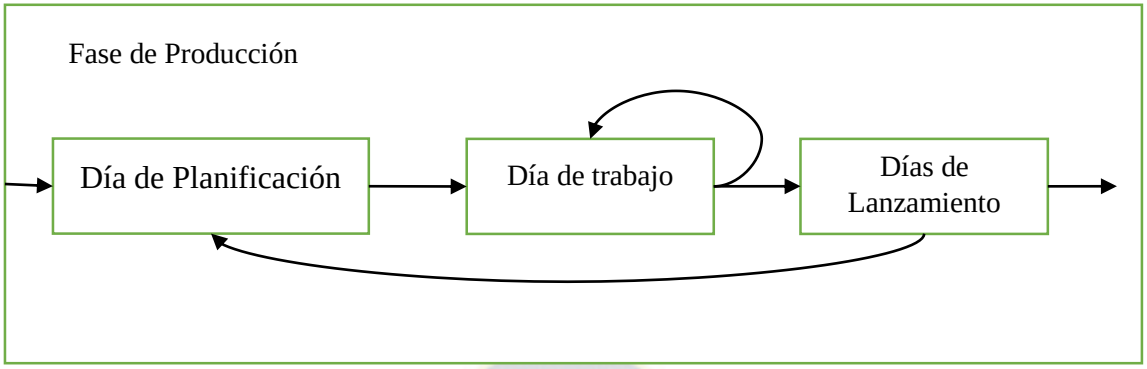


Figura 2.6 Composición de la fase de Producción
Fuente: [Elaboración Propia]

2.3.3.1. Etapas

a) Día de Planificación - Planning day.- El objetivo son los días para definir los contenidos (es decir, historias y tareas) para la iteración.

b) Dia de Trabajo - Working day.- Su objetivo es implementar la funcionalidad decidida, de una manera controlada y gestionada.

c) Día de Lanzamiento - Release day.- El propósito de la etapa es verificar y validar la funcionalidad implementada. Típicamente, el día de liberación culmina en una liberación real, pero la liberación puede ser no oficial para que el producto se evalúa sólo dentro de los interesados en el proyecto básico.

2.3.3.2. Roles

La fase Producción utiliza el mismo equipo del proyecto y los actores presentes en las fases anteriores. Sin embargo, el papel de la comunicación con el cliente debe enfatizarse como retroalimentación rápida durante la implementación, es beneficioso para el logro de resultados satisfactorios es decir, mejor correspondencia con los requisitos.

2.3.3.3. Patrones relacionados

Otros patrones que forman “parte de”, “compuestos por”, o “asociados” estrechamente a la fase de producción se identifican aquí.

a) Fase de inicialización proporciona los insumos para esta fase.
b) Fase Producción se compone de día Planificación, día de trabajo, y patrones de días de lanzamiento.
c) Estabilizar fase continúa de esta fase.

2.3.4. Estabilización

El propósito del patrón Estabilizar fase es asegurar la calidad de la implementación del proyecto.

Se llevan a cabo las acciones de integración para asegurar que el sistema completo funciona correctamente, para enganchar los posibles módulos separados en una única aplicación. Los objetivos del modelo de fase de estabilización son:

a) Finalizar la implementación del producto.
b) Mejorar y garantizar la calidad del producto.
c) Finalizar la documentación del producto.

Los criterios de ingreso:

- La fase precedente se ha completado. Esto significa que la valor esencial funcionalidad y empresarial del proyecto ha sido implementado y se muestra al cliente.

Entradas:

- La funcionalidad implementada del producto.

- Los artefactos de desarrollo relacionados.

Criterios de salida:

- Cien por ciento de la funcionalidad necesaria del proyecto está trabajando y cumple con las necesidades del cliente.

Salidas:

- Funcionalidad implementada de todo el software de proyecto.
- Documentación del producto finalizado.

El modelo de fase de Estabilización puede llevarse a cabo a través de las siguientes etapas:

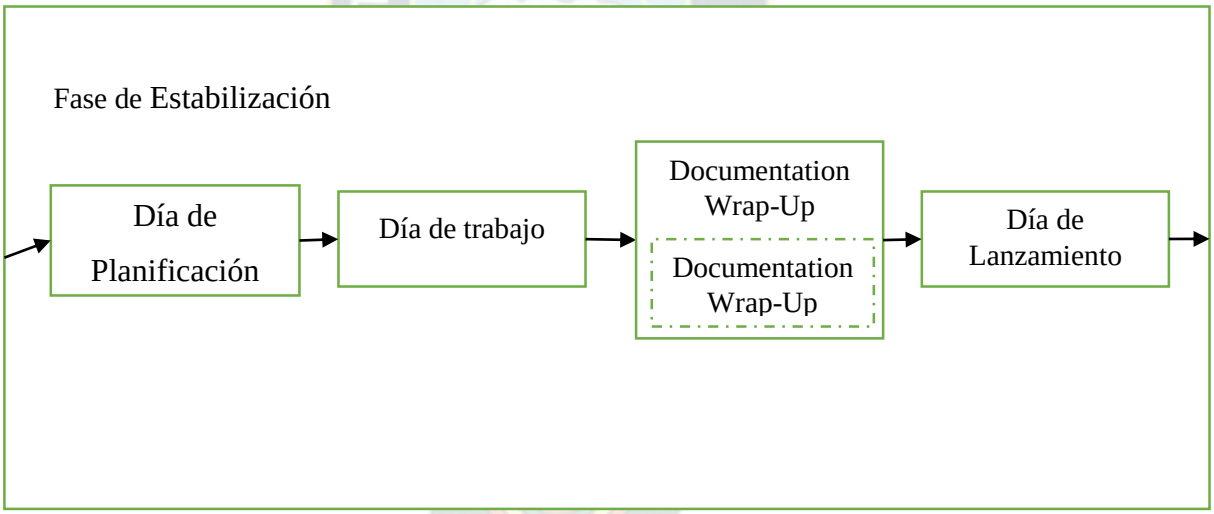


Figura 2.7. Composición de la fase de Estabilización
Fuente: [Elaboración Propia]

2.3.4.1. Etapas

a) Día de Planificación - Planning day.- El objetivo de la etapa es definir el contenido (es decir, historias y tareas) para la implementación de las características del producto y para los restantes la mejora de la calidad externa e interna del producto (re factorizando).

b) Dia de Trabajo - Working day. - El objetivo es finalizar la ejecución del producto, así como mejorar y asegurar la calidad del producto.

c) Documentación WrapUp.- El objetivo de la etapa es finalizar La arquitectura de software, diseño de interfaz de usuario y documentos como se muestra en la figura 2.5. La etapa Documentación WrapUp incluye una sola tarea que se puede llevar a cabo utilizando el patrón de tarea de la documentación WrapUp.

d) Día de Lanzamiento - Release day.- El propósito de la etapa es para verificar y validar la funcionalidad implementada y la calidad de todo el software y su documentación. El Día de Lanzamiento culmina en la versión final de todo el software.

2.3.4.3. Patrones relacionados

Otros patrones que forman "parte de", "compuestos por", o "asociados" estrechamente a la fase de Estabilización se identifican aquí.

a) El modelo de la fase de producción se utiliza para implementar y proporcionar insumos para esta fase.
b) Este modelo que utiliza el Planning Day, jornada de trabajo, documentación Wrap Up y patrones Release Day para llevar a cabo el patrón.
c) El patrón de prueba del sistema y Fix se utiliza para aplicar la prueba del sistema y Fijar fase que es la sucesora de la fase de Estabilización.

2.3.4.4. VARIACIONES

Las dos variaciones de la fase de Estabilización han sido identificadas:

a) Estabilizar a través de varios comunicados.

Sólo se necesita una liberación que se produce en una fase de Estabilización ideal.

Sin embargo, si se necesitan varios lanzamientos entonces la siguiente estabilización a través de varios comunicados de modelo se puede aplicar.

La documentación WrapUp,etapa precede sólo el lanzamiento final. Estabilizar través de varios comunicados.

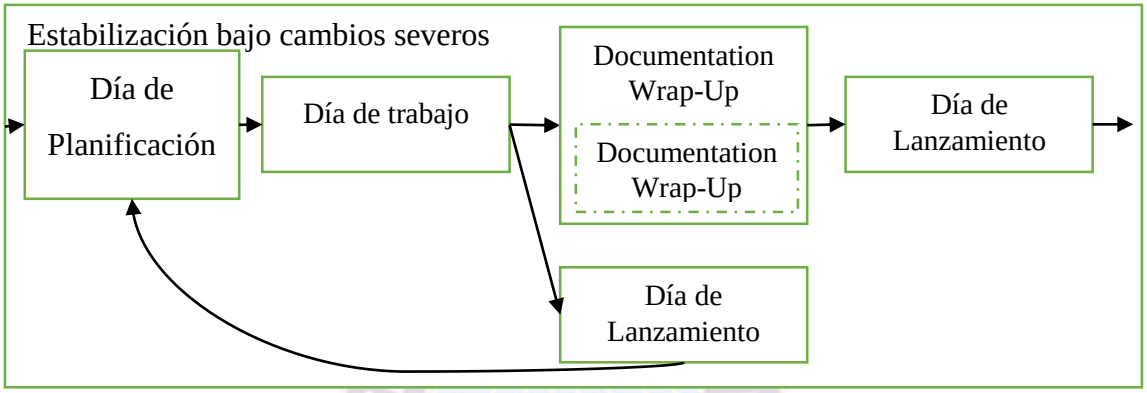


Figura 2.8 Composición de la fase de Estabilización
Fuente: [Elaboración Propia]

b) La variante de la etapa del día Planificación

La etapa del día Planificación de la fase de Estabilización puede incluir la definición del contenido (es decir, historias y tareas) para finalizar la arquitectura de software, diseño de interfaz de usuario y los documentos de la Conclusión de la fase de documentación.

2.3.5. 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 acabadas todas las fases deberíamos tener una aplicación publicable y entregable al cliente.

Los objetivos del modelo de fase de estabilización son:

a) Pruebe el sistema basado en la documentación producida en el proyecto.
b) Proporcionar información de defectos encontrados.
c) Deje al equipo de proyecto para planificar solución para los defectos encontrados.
d) Fijar los defectos.
e) Produce como sistema libre de errores como sea posible

Los criterios de ingreso:

- Las fases anteriores necesitan ser terminadas, por lo que las pruebas se pueden realizar con un sistema en funcionamiento.
- Funciones definidas para el equipo de pruebas.

Entradas:

- La funcionalidad implementada,
- Documentación de pruebas de aceptación,
- Definido por el usuario de funcionalidad.
- Descripción de la interfaz de usuario se utiliza para crear casos de prueba.

Criterios de salida:

- Los defectos identificados son fijos.
- Liberación final se compone como se discute en el patrón relacionado

Salidas:

- Una prueba y sistema fijo (la versión final).
- Documentación de los defectos encontrados.
- Informe de la prueba del sistema. Una breve descripción del proceso de prueba y la encontró defectos.

- Registro de prueba. Un documento que contiene todos los casos de prueba que se ejecutan en el sistema, prueba y resultados de las pruebas.

La siguiente figura ilustra las etapas de las cuales la Fase de Prueba del sistema y solución es compuesto de:

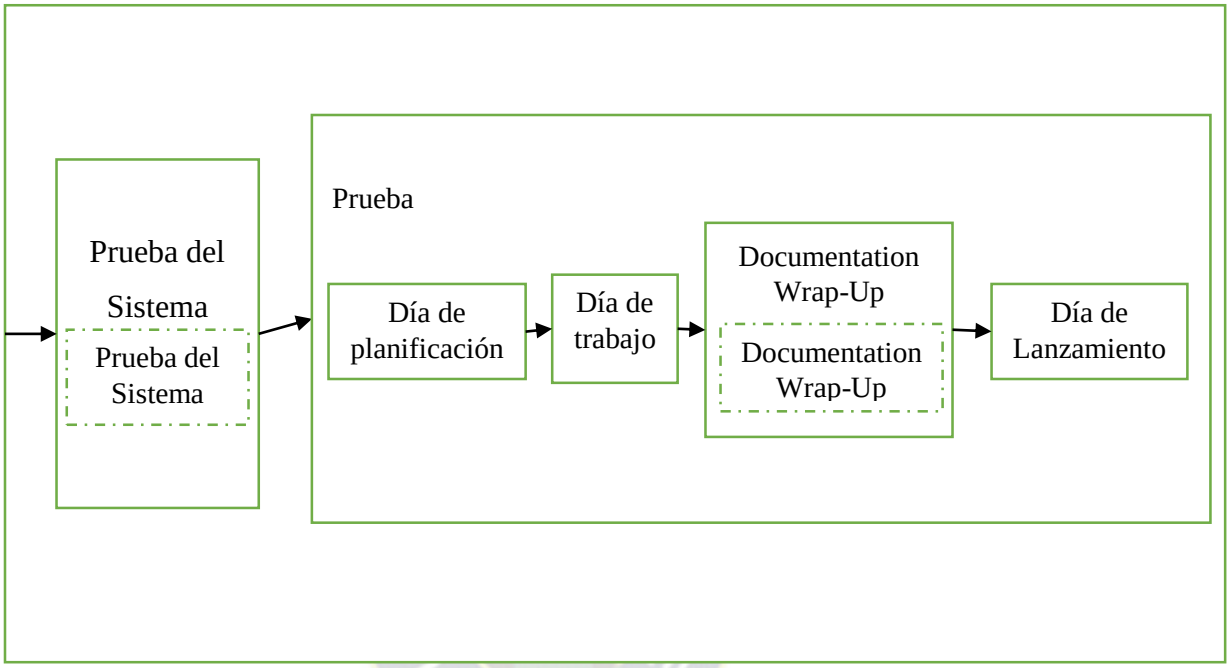


Figura 2.9 Composición de la fase de Prueba
Fuente: [Elaboración Propia]

2.3.5.1. Etapas

Las etapas individuales de la prueba del sistema y la fase pruebas son:

a) Prueba del sistema

Es una etapa en la que el sistema se prueba como se describe en la Prueba del sistema de patrón de tarea. Defectos encontrados se documentan con el propósito de la iteración Fix. Es una variación de la iteración normales; Sin embargo ninguna nueva funcionalidad es implementada y la escala de tiempo puede ser notablemente más corto. La entrada para esta iteración es los defectos encontrados en la fase de prueba del sistema.

b) Prueba

Es una variación de las iteraciones normales, sin embargo ninguna nueva funcionalidad es implementada y la escala de tiempo puede ser notablemente más corto.

La entrada para esta iteración son los defectos encontrados en la fase de prueba del sistema.

- **Día de Planificación.** El objetivo de la fase de planificación de días de la prueba del sistema Y fase Fix es definir el contenido (es decir, historias y tareas) para la corrección iteración.
- **Día de Trabajo.** El objetivo de la fase de días de trabajo del Sistema Prueba y fase de solución es arreglar los defectos encontrados en la fase de prueba del sistema y para finalizar la implementación del producto.
- **Documentación** WrapUp. El objetivo de la Documentación wrap Hasta esta etapa es para finalizar la arquitectura de software, el diseño y la interfaz de usuario documentos. La documentación se ha actualizado para que se corresponda con los cambios realizado durante la iteración Fix. Como se muestra en la figura, la Documentación WrapUp etapa incluye una sola tarea, la Tarea de documentación WrapUp, que se puede llevar a cabo utilizando la Documentación WrapUp.
- **Día de lanzamiento.** para verificar el software y su documentación. El día del lanzamiento culmina en la versión final de todo el software.

2.3.5.2. Roles

Las siguientes funciones están relacionadas con la fase de Solución:

a) Grupo de Proyecto
b) Grupo de Apoyo
c) Cliente
d) Grupo de dirección.
e) Grupo de prueba de sistema

2.4. Ingeniería Web

Las metodologías, técnicas y herramientas que se utilizan en el desarrollo de aplicaciones web complejas y de gran dimensión en las que se apoya la evaluación, diseño, desarrollo, implementación y evolución de dichas aplicaciones, hacen referencia a la ingeniería web. Entonces la aplicación de principios de ingeniería pueden poner bajo control el desarrollo de aplicaciones Web, minimizando riesgos y mejorando el mantenimiento y calidad.

2.4.1. Modelado de Lenguaje Web - WEBML

Es una metodología de modelado visual de aplicaciones web, centrada especialmente en las aplicaciones de uso intensivo de datos, separando el contenido de la información en páginas, navegación y presentación, que se pueden definir y desarrollar de forma independiente. Permite la especificación de operaciones de manipulación de datos para actualizar la aplicación.

Cuenta con cuatro perspectivas: el **Modelo de datos** (de los datos de la aplicación), **el Modelo de Hipertexto** (para cada hipertexto describe qué páginas lo componen, y cómo navegan), **el Modelo de Presentación** (disposición y apariencia gráfica), y **el Modelo de Personalización** (para definir operaciones específicas para usuarios ó grupos de usuarios, ya que se almacenan como entidades en el Modelo Estructural). Dispone de una herramienta CASE (WebRatio). Es un modelado orientado a aplicaciones con un uso intensivo de datos, donde hay gran cantidad de datos, con estructura compleja y las aplicaciones tienen que acceder a ellos.

2.4.1.1. Análisis de requerimientos

Análisis de requerimientos se centra en la recogida de información sobre el dominio de aplicación y las funciones que se esperan y sobre la especificación de ellas a través de fáciles de entender las descripciones. La entrada a esta actividad es el conjunto de los requerimientos

del negocio que motivan el desarrollo de aplicaciones. Los principales resultados de esta fase son

- La identificación de los grupos de usuarios que aborda la aplicación. Cada grupo representa a los usuarios que tengan las mismas características o jugando el mismo papel dentro de un proceso de negocio, es decir, la realización de las mismas actividades con los mismos derechos de acceso a través de los mismos objetos. Lo mismo para el usuario individual puede desempeñar diferentes roles, que pertenece así a diferentes grupos.
- La especificación de requisitos funcionales que se ocupan de las funciones que deben facilitarse a los usuarios. Para cada grupo de usuarios, las actividades relacionadas que deben realizar se identifican y especifican.
- La identificación de los objetos de información básicos, es decir, la información principal activos para acceder, intercambio, y / o manipulados por los usuarios.
- La descomposición de la aplicación web en vistas de sitio, es decir, diferente hipertextos diseñados para satisfacer un conjunto bien definido de funcional y de usuario requisitos. Cada grupo de usuarios estará provisto de al menos un sitio ver apoyar las funciones identificadas para el grupo.

Se espera que los analistas de utilizar su formato favorito para los requisitos especificación por ejemplo, formatos tabulares pueden ser utilizados para la captura de los requisitos informales, como grupo o ver sitio descripciones, Uso UML diagramas de casos y los diagramas de actividad se pueden utilizar también como estándar y representaciones de escenarios de uso y la sincronización de la actividad. En particular, requisitos funcionales pueden ser capturadas por el flujo de actividad, mostrando secuencia, y el paralelismo y la sincronización entre las actividades a ser realizada por diferentes grupos de usuarios.

2.4.1.2. Modelado de datos

Diseño de datos es una de las disciplinas más tradicionales y consolidadas de tecnología de la información, para lo cual bien establecida lenguajes de modelado y existen directrices. Por esta razón, WebML no propone aún otros datos lenguaje de modelado, sino que explota el modelo de datos entidad-relación, o el subconjunto equivalente de UML primitivas diagrama de clases. Los elementos fundamentales del modelo de datos WebML son por lo tanto entidades, definidas como contenedores de elementos de datos, y las relaciones, definidas como semántica, conexiones entre entidades. Las entidades han nombrado propiedades, llamados atributos, con un tipo asociado.

Las entidades pueden organizarse en jerarquías de generalización y las relaciones pueden ser restringidas por medio de restricciones de cardinalidad.

En el diseño de aplicaciones Web a menudo es necesario para calcular el valor de algunos atributos o relaciones de una entidad a partir del valor de algunos otros elementos del esquema. Atributos y relaciones así obtenidas son llamadas derivadas.

Atributos y relaciones derivados pueden ser denotados añadiendo un carácter de barra (/) delante de su nombre, y su regla de cálculo pueden ser especificadas como una expresión lógica añadido a la declaración del atributo o relación, como es habitual en los diagramas de clase UML

El modelo de datos representa las diferentes tablas de datos y sus relaciones que son necesarias para una aplicación Web concreta. Se pueden utilizar:

- Diagramas de Entidad-Relación (E-R) que muestran todas las tablas, los diferentes campos de cada tabla, y las relaciones entre ellas.
- Diagramas UML de clases que pueden representar la misma información que un diagrama de Entidad-Relación (por lo que puede usarse de manera equivalente), e incluso información adicional sobre el modelo de datos.

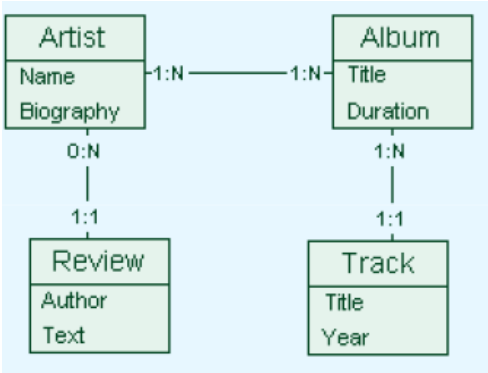


Figura 2.10 Modelado Estructural
Fuente: [Webml, 2014]

2.4.1.3. Modelado de hipertexto

El modelo de hipertexto permite la definición de la interfaz de front-end, que se muestra a un usuario en el navegador. Permite la definición de páginas y su organización interna en términos de componentes (llamados unidades de contenido) para mostrar el contenido. También es compatible con la definición de enlaces entre páginas y unidades de contenido que soportan la ubicación de la información y la navegación. Los componentes También puede especificar las operaciones, tales como la gestión de contenidos o usuario de procedimientos de inicio de sesión / cierre de sesión. Estos se llaman unidades de operación.

La estructura modular de un extremo frontal de aplicación se define en términos de vistas del sitio, áreas, páginas y unidades de contenido. Una vista es un especial sitio hipertexto, diseñado para hacer frente a un conjunto específico de requisitos. Consiste en áreas, que son las principales secciones del hipertexto y comprende recursivamente otras sub áreas o páginas. Las páginas son los contenedores reales de información entregada al usuario.

Varias vistas del sitio pueden ser definidos en la parte superior del mismo esquema de datos, por atender las necesidades de las diferentes comunidades de usuarios o para organizar el contenido como solicitado por diferentes dispositivos de acceso como PDAs, teléfonos inteligentes, y similares electrodomésticos.

Soporta la definición de links entre páginas y contenidos de unidades de búsqueda e información de locación.

Cada hipertexto describe una vista del sitio

- Modelo de composición. Representa las páginas de un hipertexto y cada página que elementos de contenido tiene.
- Modelo de navegación. Representa los enlaces entre las diferentes páginas y sus elementos de contenido.

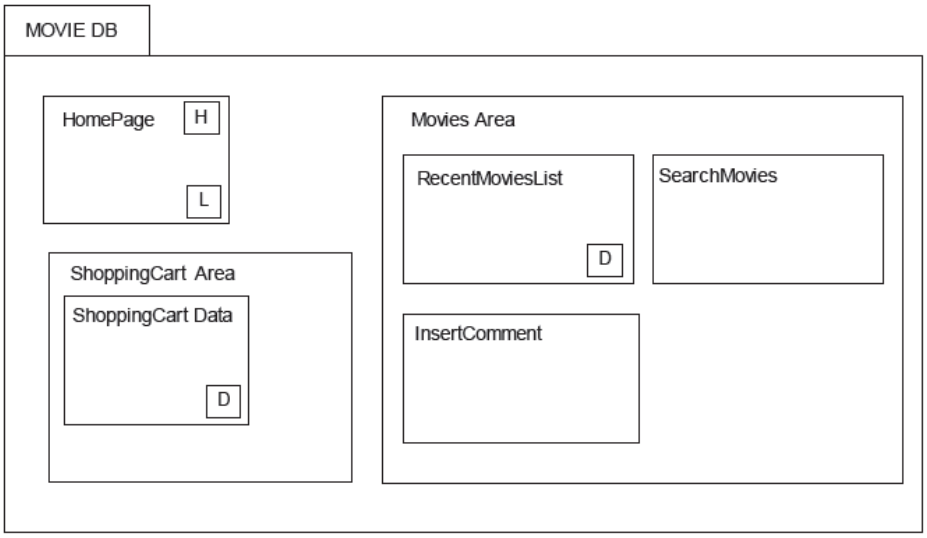
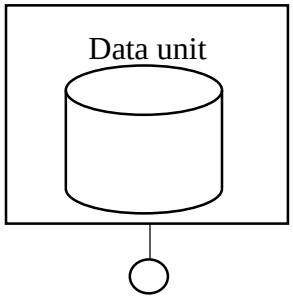
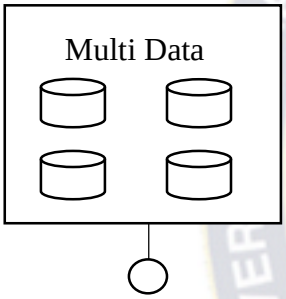
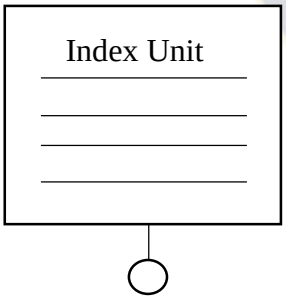


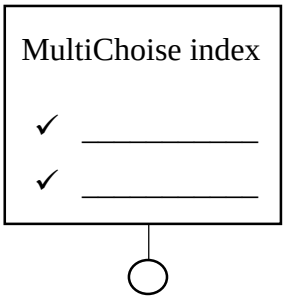
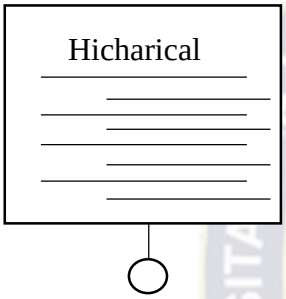
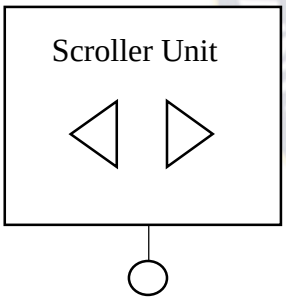
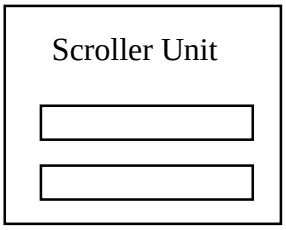
Figura 2.11 Modelado de Hipertexto
Fuente: [Webml, 2015]

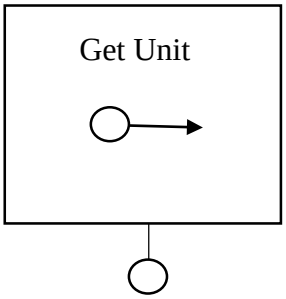
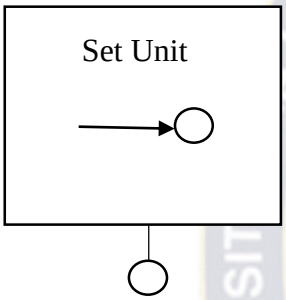
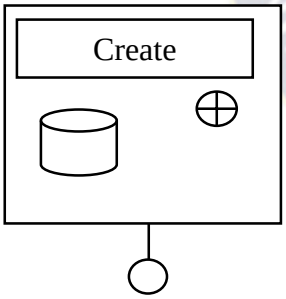
a) Modelo de composición

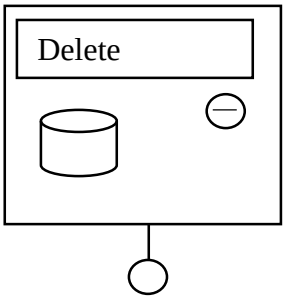
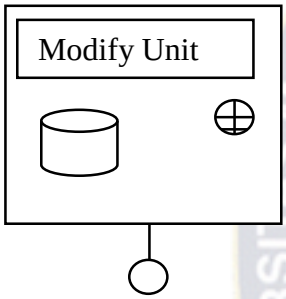
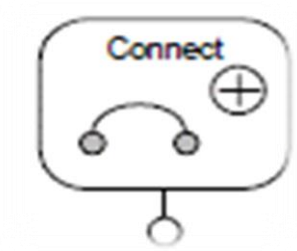
Elementos de contenido de cada página:

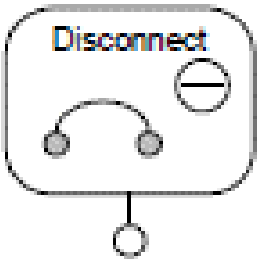
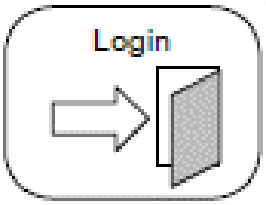
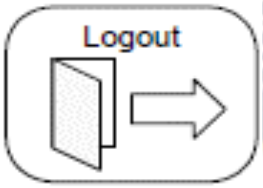
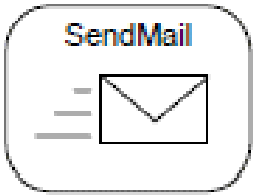
ELEMENTOS WEBML	DESCRIPCIÓN
Data Unit	Una unidad de datos publica un solo objeto de una determinada entidad

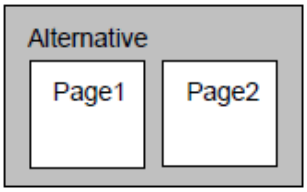
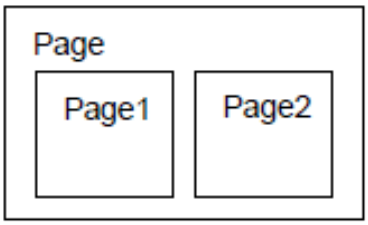
	
Multo Data Unit 	Presenta múltiples objetos y/o unidades de una entidad en conjunto.
Index Unit 	Presenta una lista de varios objetos de una entidad.
Multi Choise – Index	Permite al usuario la selección de múltiples objetos de una lista asociados a una casilla.

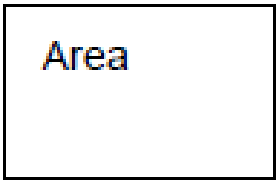
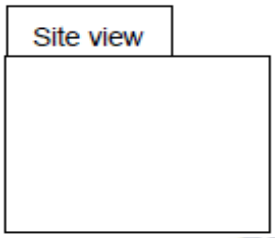
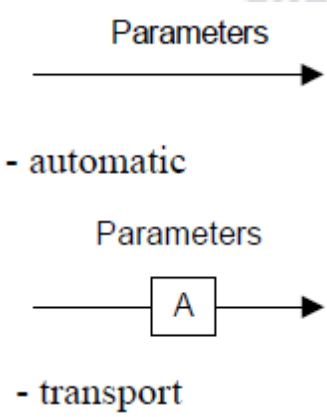
	
Hierachical Index Unit 	Organiza las entradas de índices en un árbol de múltiples niveles.
Scroller Unit 	Proporciona comandos para desplazarse través de los objetos en un conjunto.
Entry Unit 	Admite la entrada de datos basada en formularios.
Get Unit	

	Retorna el valor del parámetro global
Set Unit 	Asigna un valor a un parámetro global
Create Unit 	Permite la creación de una nueva instancia de entidad.
Delete Unit	Borra uno o más objetos de una entidad.

	
Modify Unit 	Actualiza uno o más objetos de una entidad.
Connect Unit 	Crea nuevas instancias de una relación
Disconnect Unit	Elimina la instancia de una relación

	
Login Unit 	Verifica la identidad de un usuario al ingresar al sitio.
Logon Unit 	Envía al usuario a la página predeterminada si control de acceso.
SendMail Unit 	Proporciona la capacidad para el envío de mensajes

Generic Operation Unit 	Define una operación genérica: los parámetros de entrada y salida deben ser definidos por el diseñador.
Transaction 	Es una secuencia de operaciones ejecutadas automáticamente.
Page 	Representa la interfaz actual navegada por el usuario. Contiene unidades y/o sub-páginas.
OR Sub – Pages 	Sub-páginas OR especifican que ciertas partes de la pantalla pueden contener piezas alternativas de contenido, cada una se modela como una página distinta.
AND Sub – Pages 	Sub-páginas AND se utilizan para dividir la página contenida en una pantalla en porciones.

Área  	Es un contenedor de páginas o, de forma recursiva, otras sub-áreas, lo que puede ser usado para dar una organización jerárquica del hipertexto.
Site View  	Representa un Hipertexto
Link  	Un enlace es una conexión orientada entre dos unidades o páginas. Permite el transporte de información entre unidades. Pueden ser definidos como: - Automáticos: son navegados sin la intervención del usuario. - Transporte: Sólo permiten la transmisión de parámetros

Parameters - OK link Parameters OK - KO link Parameters KO	Enlaces de operaciones de salida se distinguen en: - Enlace OK: se trata de seguir en caso de operación éxito - Enlace KO: que se siguen en caso de un fracaso de operación.

Tabla 2.1. Modelos de diseño de hipertexto de WEBML
Fuente: [Elaboración propia]

b) Modelo de Navegación

Tipos de enlaces

i) No contextuales. Conectan semánticamente páginas independientes. Elemento HYPERLINK.
ii) Contextuales. La página del destino depende de la del origen de dicho enlace, en el sentido del esquema de datos definido. Elemento INFOLINK

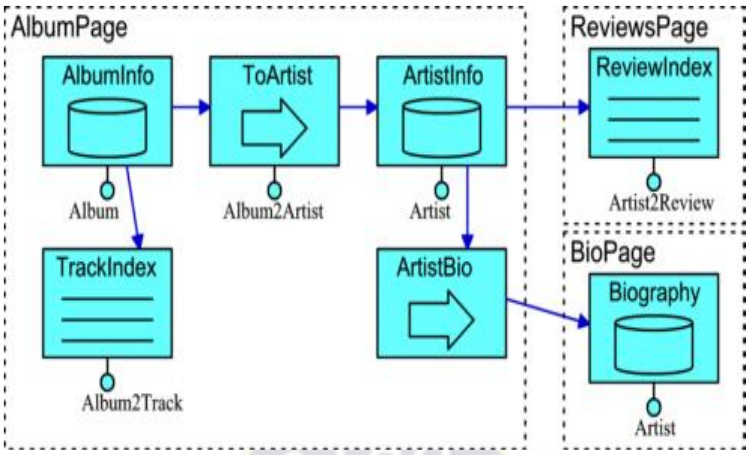


Figura 2.12 Modelado de Hipertexto
Fuente: [Webml, 2015]

c) Modelado de gestión de Contenido o Personalización

Describe la extensión del modelo de hipertexto para apoyar las funciones de gestión de contenidos, como la actualización de la información personal, el llenado de carros, y así sucesivamente. Nuevas construcciones se introdujeron para las operaciones que representan, que son ya sea predefinido o genérico. Operaciones predefinidas representan las funciones de gestión de contenidos y de servicios públicos típicos que normalmente se encuentran en los sitios web, como la creación, supresión y modificación de objetos, de inicio de sesión del usuario y cierre de sesión, y la entrega de mensajes de correo electrónico; operaciones genéricos representan funciones de recuadro negro y permiten la integración de aplicaciones WebML con servicios externos.

2.5. Cloud computing

La computación en la Nube o Cloud Computing, consiste en la evolución de varios conceptos relacionados con las tecnologías de la información, como ser la virtualización, diseño de redes, diseño de aplicaciones distribuidas. Es la respuesta a las necesidades de los departamentos de tecnología de información de las empresas en la búsqueda de mayor escalabilidad, alta disponibilidad y reducción de los costes operativos.

Las características fundamentales que debe proporcionar un servicio para poder ser considerado “Cloud Computing” son:

- Auto servicio por parte de los clientes
- Capacidad de medir el servicio
- Pago por uso
- Rápida elasticidad
- Distribución geográfica.

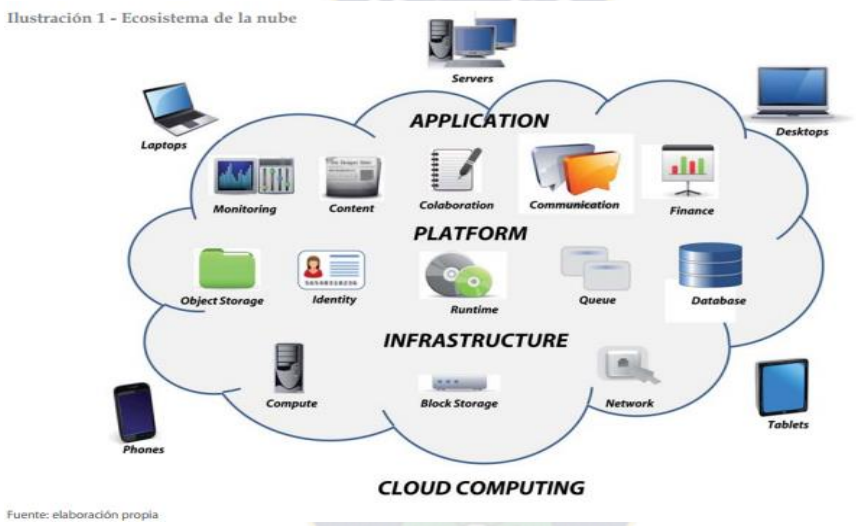


Figura 2.13. Cloud Computing
Fuente: [Management studios, 2015]

2.6. SAAS (Software como Servicio)

Software como un Servicio. Abreviadamente SaaS (del inglés: Software as a Service, SaaS) es un modelo de distribución de software donde el soporte lógico y los datos que maneja se alojan en servidores de una compañía de tecnologías de información y comunicación (TIC), a los que se accede vía Internet desde un cliente. La empresa proveedora TIC se ocupa del servicio de mantenimiento, de la operación diaria y del soporte del software usado por el cliente. Regularmente el software puede ser consultado en cualquier computador, se encuentre presente en la empresa o no. Se deduce que la información, el procesamiento, los

insumos, y los resultados de la lógica de negocio del software, están hospedados en la compañía de TIC.

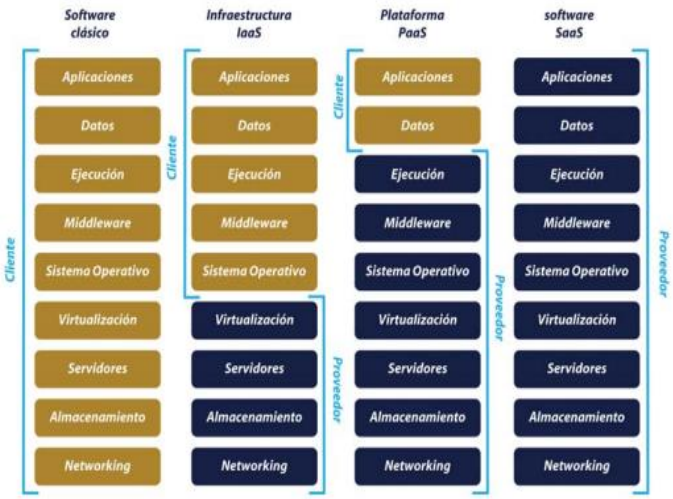


Figura 2.14. Modelos de Servicio

Fuente: [management studios,2014]

2.7. Windows Azure

Windows Azure es el sistema operativo en la nube de Microsoft. Proporciona un entorno gestionado para la ejecución y el despliegue de aplicaciones y servicios en la nube. Windows Azure proporciona a los desarrolladores un entorno de computación bajo demanda y almacenamiento alojado en los centros de datos de Microsoft para aplicaciones en la web.

Aunque Windows Azure permite reutilizar todos los conocimientos de .NET es también una plataforma abierta a otros lenguajes y plataformas mediante el uso de estándares y el soporte para lenguajes ajenos a la plataforma .Net como, por ejemplo, PHP, lenguajes no manejados como C/C++ nativo, así como soporte para el protocolo FastCGI.

Desde el punto de vista del desarrollo con .NET, Windows Azure permite ejecutar aplicaciones ASP.NET y código .NET en la nube, proporcionando una plataforma de ejecución basada, a día de hoy, en el framework de .NET 4.0 e IIS 7, complementado con un

API de almacenamiento y de "tracing" propios de la plataforma Azure. Ofrece un portal que permite gestionar las aplicaciones Azure de una manera cómoda y natural.

El entorno de ejecución de Windows Azure es 'full trust' lo que permite ejecutar servicios de Windows Communication Foundation e incluso llamar a código nativo desde las aplicaciones Azure basadas en .NET.

Además Windows Azure va a proporcionar servicios de almacenamiento no relacional y colas con acceso autenticado, alta redundancia (triple) y accesible mediante una interfaz REST que se puede utilizar desde cualquier lenguaje que tenga la capacidad de realizar peticiones HTTP, que es tanto como decir cualquier lenguaje moderno.

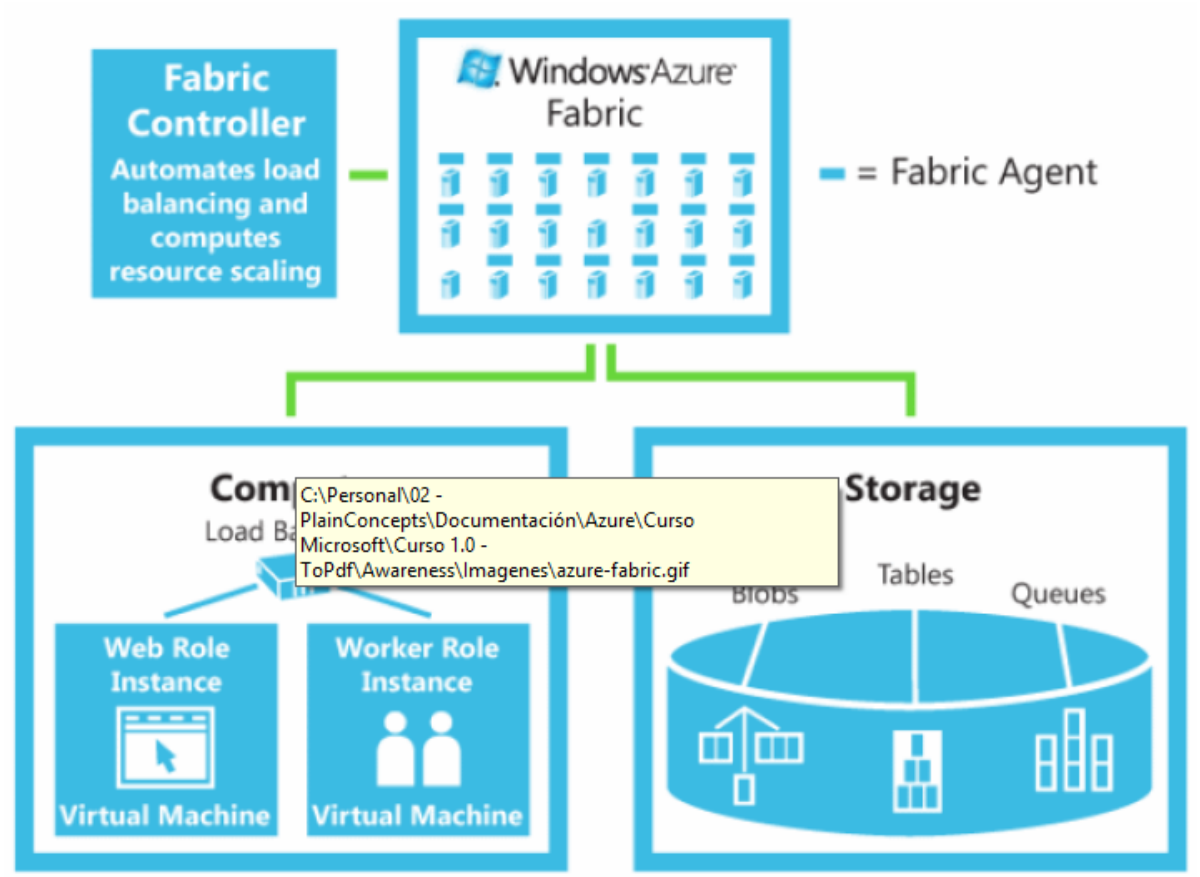


Figura 2.15 Arquitectura de la infraestructura de Windows Azure
Fuente: [Landa, 2011]

2.8. Georreferenciación

La georreferenciación es la técnica de posicionamiento espacial de una entidad en una localización geográfica única y bien definida en un sistema de coordenadas y datos específicos. La georreferenciación es un aspecto fundamental en el análisis de datos geoespaciales, pues es la base para la correcta localización de la información de mapa y, por ende, de la adecuada fusión y comparación de datos procedentes de diferentes sensores en diferentes localizaciones espaciales y temporales. Por ejemplo, dos entidades georreferenciadas en sistemas de coordenadas diferentes pueden ser combinables tras una apropiada transformación afín (bien al sistema de coordenadas del primer objeto, bien al del segundo).

2.8.1. Sistemas de Coordenadas y Datum Geodésicos

Los sistemas de coordenadas pueden desglosarse en dos grandes grupos:

a) Sistemas de coordenadas geográficas, si la entidad es descrita en términos de coordenadas latitud-longitud asociada a un datum geodésico específico.
b) Sistemas de coordenadas proyectadas, si son coordenadas referidas a un plano en el cual se ha proyectado parte de la superficie terrestre modelada con un datum. Dado que no es posible una proyección sin distorsión entre una superficie elipsoidal y un plano, estos sistemas de coordenadas se restringen a regiones suficientemente pequeñas para minimizar estos efectos.

2.8.2. Métodos de Georreferenciación

Se pueden distinguir dos métodos principales de georreferenciación:

2.8.2.1. La georreferenciación Orbital

En la que se modelan las fuentes de error geométrico conocidas (la curvatura terrestre, la distorsión panorámica, la rotación terrestre, etc.) y se aplican transformaciones inversas que

corrijan estos errores intrínsecos y sistemáticos de forma automatizada. Tiene la principal ventaja de que no necesita intervención humana una vez que es implementado, pero puede dar lugar a grandes errores en las coordenadas de las imágenes de satélite si su sistema de posicionamiento no tiene la suficiente precisión (problema que ha disminuido con la llegada de los sistemas de navegación modernos).

2.8.2.2 La Georreferenciación por Puntos de Control

En la que a partir de un conjunto de puntos bien identificados en la imagen y de los que se conocen sus coordenadas se calculan las funciones de transformación (lineales, cuadráticas) que mejor se ajustan a estos puntos. Para que esta georreferenciación resulte satisfactoria es necesario elegir de forma apropiada los puntos de control (en número, ubicación y distribución). Se trata, pues, de un proceso manual en el que se requiere intervención humana. Ofrece mayor exactitud cuándo se trabaja en zonas donde es posible identificar bien los puntos conocidos.

2.9. Prevención de Criminalidad Urbana

Las causas de la delincuencia son múltiples, son una serie de causas interrelacionadas. Se pueden agrupar tres categorías: sociales, institucionales y las que se refieren al entorno urbano y físico. La delincuencia urbana ha llegado a constituir un problema muy serio de debido a políticas de prevención.

Probablemente las historias de prevención y a menudo en apoyo a estas han surgido muchas propuestas de apoyo policial que apuntan a La creación de fuerzas especializadas, claras coherentes y estables. La impuntualidad policial frente a la sociedad civil y sus autoridades, el trabajo en estrecha colaboración con las coaliciones locales de seguridad.

CAPÍTULO III

MARCO APLICATIVO

3.1. Introducción

El presente capítulo tiene como finalidad describir el análisis y diseño del software como servicio para la prevención de criminalidad urbana utilizando métodos de georreferenciación, utilizando la metodología Mobile – D relacionada a XP dentro de las Iteraciones, historias de usuario, y puntos en los que nos ayudara a detallar el tiempo y las actividades realizadas para lograr nuestro objetivo. A continuación se muestra el desarrollo Mobile – D y Webml detallando sus respectivas acciones de implementación del software como servicio denominado "FEELSAFE"

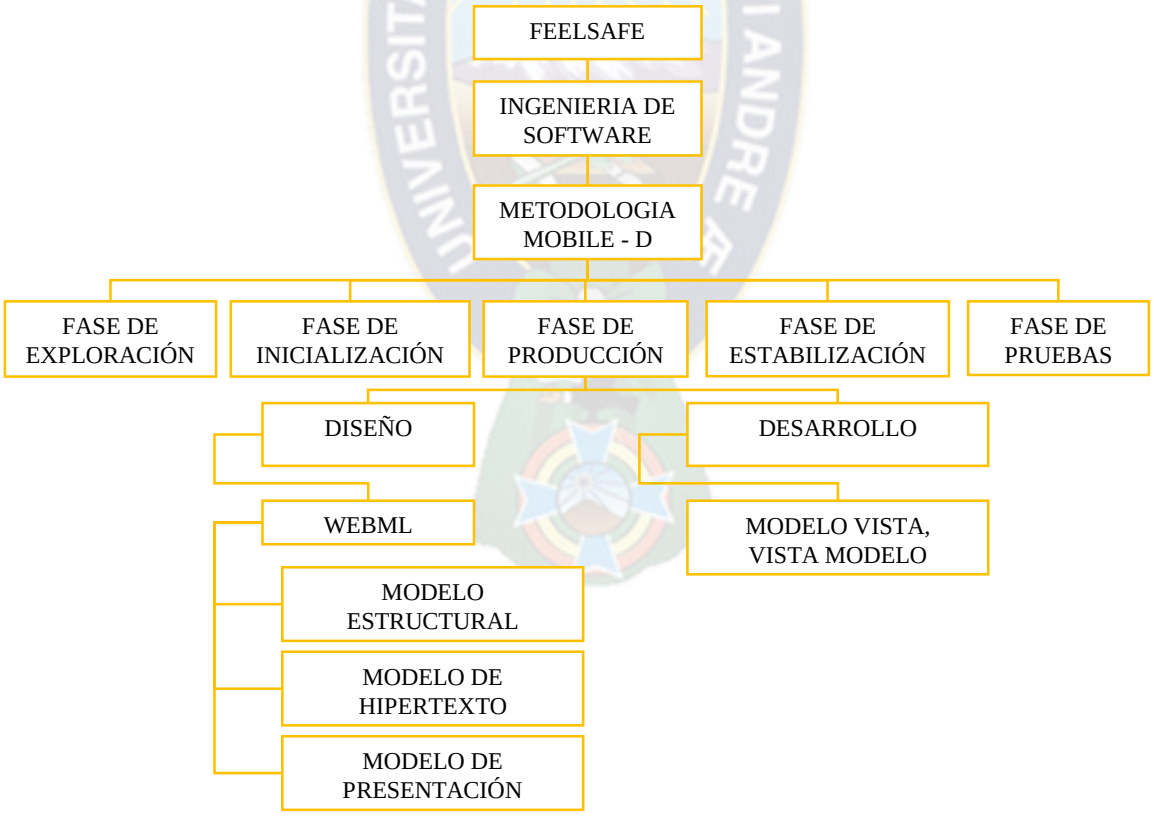


Figura 3.1: Planificación de la Implementación del Software como servicio FEELSAFE
Fuente: [Elaboración propia]

El modelado Web (Webml), se tomará en cuenta por ser un proyecto que utiliza recursos en la nube (cloud computing) y Mobile Services. Este modelo nos ayudara a obtener un entorno de aplicación más amigable para el usuario obteniendo, el resultado esperado junto a las métricas deseadas.

3.2. Fase de exploración

El Propósito de la fase de exploración es la planificación y establecimiento del proyecto, dando las bases de la aplicación, para que de esta manera poder implementar el software y así establecer las bases.

Los objetivos de esta fase son: Definir el establecimiento de StakeHolders (Grupos de interés), definición del alcance que tendrá el proyecto con respecto al desarrollo del software como servicio, identificar los requerimientos iniciales, análisis sobre los requisitos del sistema.

3.2.1. Establecimiento del proyecto

El escenario que se quiere aplicar es tener información sobre criminalidad urbana, información de los lugares peligrosos a partir de la valoración que se dé a cierta zona. Detallar la posibilidad de riesgo a determinada hora del día y posición GPS del usuario en un mapa georreferenciado. Todo esto dentro de una aplicación en el dispositivo móvil, ya que de esta manera se disminuye costos y aumenta la movilidad y flexibilidad para el monitoreo.

Facilitar al usuario una aplicación para su conocimiento en zonas delincuenciales en diferentes puntos de la ciudad en donde está. De esta manera hacer una demostración preventiva ante situaciones de criminalidad, y delincuencia al llegar a un determinado lugar.

Para comenzar necesitamos encontrar la manera de comunicar toda la información obtenida y por obtener con el dispositivo Windows Phone, por lo cual está planificado sincronizarlo con la ayuda de Windows Azure, utilizando las bases de datos en la Nube.

Una vez realizada la identificación y sincronización se hará una llamada a la obtención de interfaz detallada al usuario para su interacción con la aplicación.

3.2.2 Establecimiento de Grupos de Interes Stakeholders

El propósito de esta tarea es establecer los StakeHolders (grupos de interés):

- Usuarios que a través de la aplicación podrán determinar lugares georreferenciados para la prevención de criminalidad. A través de la información obtenida en el servicio móvil. (Mobile Service).
- Personas interesadas en saber las estadísticas de delincuencia en la ciudad.
- Público en General.

3.2.2. Definición de alcance

Se ha establecido realizar un diagrama que muestre una síntesis de lo que se quiere realizar y a donde se quiere llegar con el desarrollo del proyecto.

Cronograma de Avance del Proyecto																											
Actividades	Duracion en Días	Del 1 de Julio al 30 de Noviembre																									
		Julio				Agosto				Septiembre				Octubre				Noviembre				Diciembre					
Fase de Exploración	30 Días																										
Fase de Inicialización	22 Días																										
Fase de Producción	35 Días																										
Fase de Estabilización	21 Días																										
Fase de Pruebas	30 Días																										

Tabla 3.1.: Cronograma de Avance del Proyecto
Fuente: [Elaboración propia]

3.1.2.1 Análisis sobre requisitos del sistema

En este segmento se listaran los requisitos que se deben cumplir para hacer un buen uso de la aplicación.

- Se deberá contar con un dispositivo móvil con sistema operativo Windows Phone 8.1 o superior.
- Se necesita una red inalámbrica WiFi y acceso a internet en el dispositivo móvil.
- La aplicación será de uso común, por lo que cualquier persona tendrá acceso a esta.
- Se minimizaran los errores que se puedan producir a lo largo de la georreferenciación.
- La aplicación verificara datos estadísticos y mostrara índices de porcentajes para la comprensión del usuario.

3.3. Fase de Inicialización

El propósito de esta fase es permitir el éxito de las siguientes fases mediante la elaboración y verificación de los requisitos seleccionados por el cliente. En ese caso tenemos dos tipos de usuario Administrador y usuario público

3.3.1. Requerimientos Iniciales

Con los respectivos estudios de acuerdo a la definición del software como servicio se identificaron los requerimientos iniciales.

- Diseño de la Base de Datos.
- Diseño de la Interfaz de Usuario.
- Vista de Mapa y sus características.
- Referencia de probabilidad delincuencial acuerdo a la hora.
- Historial delincuencial.
- Listado de Zonas y Calles evaluadas.
- Conexión con el servidor.

3.3.2. Análisis de requerimientos iniciales

Los requerimientos iniciales identificados son los siguientes:

REQUERIMIENTO	IMPORTANCIA
Diseño de la Base de Datos	Alta
Diseño de la Interfaz de Usuario	Alta
Vista de Mapa y sus características	Alta
Referencia de probabilidad delincuencial de acuerdo a la hora.	Alta
Historial delincuencial.	Alta
Reporte de zonas y calles evaluadas	Alta
Conexión con el servidor	Alta

Tabla 3.2.: Requerimientos Iniciales Identificados
Fuente: Elaboración Propia

3.3.3. Módulos de la Aplicación

Módulo de Diseño de la Base de Datos

- Crear la Base de Datos en la Nube.
- Verificación de Tablas
- Llenado inicial de datos.

Módulo de Conexión con el servidor

- Crear conexión.
- Actualización de conexión
- Consulta de conexión con el servidor.

Módulo de Vista de Mapa y sus características

- Cargar mapa
- Registro de Zonas
- Establecer ubicación

Módulo de Referencia de probabilidad delincuencial de acuerdo a la hora

- Comparar la Hora con los datos obtenidos.
- Mostrar en el mapa actos delictivos en respectivas horas.
- Registro de información por horas especificas

Módulo de Historial delincuencial.

- Cargar datos de lugares seleccionados.
- Mostrar actos delictivos en respectivas áreas y su descripción.
- Registro de Valoración por zona.

Módulo de Reporte de zonas y calles evaluadas

- Inserción de datos de usuario
- Mostrar formulario para llenado de posible delito.
- Verificación de usuario.
- Registro de reporte de usuario.

3.3.4. Planificación inicial

En esta planificación crearemos las historias previstas para el desarrollo del proyecto, todas las historias de usuario son cambiadas a lo largo del proyecto de acuerdo al FeedBack del usuario. Se ha establecido utilizar historias de usuario de la metodología XP para la clara definición del alcance del proyecto, las iteraciones y como guía para el transcurso del proyecto.

3.3.2.1. Historias de usuario

Lo principal de un Software como servicio es el manejo detallado de la Base de datos está realizada en la Nube.

Historia de Usuario	**Número 1**	**FEELSAFE**
Nombre: Diseño de Base de Datos		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia : Se debe tener rápido y fácil acceso a la información de lugares con alto índice de criminalidad		
Objetivo: Diseñar una Base de datos que facilite el acceso a toda la información sobre criminalidad.		

Tabla 3.3.: Historia de Usuario Uno
Fuente: Elaboración Propia

La conexión con el servidor es una parte indispensable para el desarrollo de la aplicación debido a la participación de Microsoft Azure dentro del proyecto como la conexión a la base de datos.

Historia de Usuario	**Número 2**	**FEELSAFE**
Nombre: Conexión con el servidor		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia: Los datos contenidos en la base de datos estarán en la Nube para su fácil acceso de este modo se debe establecer una conexión segura.		
Objetivo: Establecer la conexión con Microsoft Azure para el uso de la base de datos en la Nube.		

Tabla 3.4.: Historia de Usuario dos
Fuente: Elaboración Propia

La presentación del a interfaz es indispensable para un buen manejo de información al interactuar con el Usuario.

Historia de Usuario	**Número 3**	**FEELSAFE**
Nombre: Diseño de la Interfaz de Usuario		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia : En la actualidad los usuarios están acostumbrados a interactuar con entornos amigables y fáciles de manejar		
Objetivo: Diseñar un entorno gráfico amigable de fácil manejo para los usuarios.		

Tabla 3.5.: Historia de Usuario tres
Fuente: Elaboración Propia

El mapa en la aplicación definirá la posición del usuario mediante el GPS y podrá mostrar el rango de criminalidad en determinada sección.

Historia de Usuario	**Número 4**	**FEELSAFE**
Nombre: Vista de mapa y sus características		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia: Se debe tener al acceso al mapa donde se encuentre el usuario mediante su ubicación por GPS.		
Objetivo: Diseñar una interfaz gráfica que coincida con el mapa y referencie puntos establecidos.		

Tabla 3.6.: Historia de Usuario cuatro
Fuente: Elaboración Propia

La referencia de acuerdo a la hora es un planteamiento nuevo dentro de las aplicaciones ya que según esta denotara ciertos acontecimientos en determinada zona permitiendo saber a detalle las horas de más peligro además del lugar.

Historia de Usuario	Número 5	FEELSAFE
Nombre: Referencia de probabilidad delincuencial de acuerdo a la hora		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia: Los datos referenciados en la aplicación tendrán referencia a determinada hora relacionando los reportes con la hora actual en la que este el usuario.		
Objetivo: Establecer dentro de la interfaz un reloj y a partir de este generar los reportes en la aplicación.		

Tabla 3.7.: Historia de Usuario cinco
Fuente: Elaboración Propia

Los historiales delincuenciales son de alta importancia en el proyecto porque de esta forma se verán datos en el mapa que determinen si la zona es segura o no esto en función del usuario y la información que dé al software.

Historia de Usuario	Número 6	FEELSAFE
Nombre: Historial delincuencial		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia: De acuerdo a un tiempo establecido se actualizarán los niveles estadísticos de los reportes sobre criminalidad con ayuda de los usuarios, en la aplicación.		
Objetivo: Diseñar una interfaz gráfica que coincida con el mapa y referencie puntos establecidos sobre crímenes efectuados y evaluados por un color referenciado de peligro para la respectiva denuncia del usuario.		

Tabla 3.8.: Historia de Usuario seis
Fuente: Elaboración Propia

De acuerdo a un color establecido de peligro, el usuario identificara mejor los lugares de riesgo en determinada zona y observara su rango de criminalidad.

Historia de Usuario	Número 7	FEELSAFE
Nombre: Reporte de zonas y calles evaluadas.		**Prioridad:** Alta
Usuario: Todos		**Iteración asignada**
Historia: Los datos contenidos en la base de datos se evaluaran para mostrar en la aplicación un rango de colores respecto a la criminalidad.		
Objetivo: Establecer dentro de la Interfaz reportes de zonas peligrosas con determinado rango		

Tabla 3.9.: Historia de Usuario siete

Fuente: Elaboración Propia

3.3.5. Planificación de Fases

FASE	ITERACIÓN	DESCRIPCIÓN
Inicialización	Iteración 0	Establecimiento del proyecto, establecimiento, análisis de requerimientos iniciales
Producción	Iteración Módulo de Diseño de la Base de Datos	Implementación del módulo de Diseño de la base de datos. Refinamiento y actualización de historias de usuario, refinamiento de las tablas.
	Iteración Módulo de Conexión con el servidor	Implementación del módulo de conexión. Refinamiento y actualización de historias de usuario, refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.

	Iteración Módulo Diseño de la Interfaz de Usuario	Implementación del módulo de Diseño de interfaz de usuario. Refinamiento y actualización de historias de usuario, refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración Módulo de Vista de Mapa y sus características	Implementación de Georreferenciación del usuario utilizando el GPS, refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración Módulo de Reporte de zonas y calles evaluadas.	Llenado de Información referencial a hechos delictivos dentro de la base de datos en la Nube, refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración Módulo de Referencia de probabilidad delincuencial de acuerdo a la hora.	Llenado de Información referencial a hechos delictivos en determinada hora dentro de la base de datos en la Nube, refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
	Iteración Módulo Historial delincuencial.	Implementación de Formularios declarativos para el uso del usuario, refinamiento de interfaces. Generación y ejecución de pruebas de aceptación.
Estabilización	Iteración Módulo de Diseño de la Base de Datos	Refactorización del módulo de diseño de la base de datos. Refinamiento y actualización historias de usuario, Refinamiento de tablas.
	Iteración Módulo de Conexión con el servidor	Refactorización del módulo de conexión con el servidor. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.
	Iteración Módulo Diseño de la Interfaz de Usuario	Refactorización del módulo de diseño de interfaz de usuario. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.

	Iteración Módulo de Vista de Mapa y sus características	Refactorización del módulo de Vista de Mapa. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.
	Iteración Módulo de Reporte de zonas y calles evaluadas.	Refactorización del módulo Reporte de zonas y calles evaluadas. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.
	Iteración Módulo de Referencia de probabilidad delincuencial de acuerdo a la hora.	Refactorización del módulo de referencia de acuerdo a la hora. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.
	Iteración Módulo Historial delincuencial.	Refactorización del módulo de conexión con el servidor. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.
Pruebas del Sistema	Iteración Pruebas del Sistema	Refactorización del módulo de conexión con el servidor. Refinamiento y actualización historias de usuario, Refinamiento de interfaces.

Tabla 3.10.: Requerimientos Iniciales Identificados
Fuente: Elaboración Propia

3.3.6. Plan de Iteraciones

Partiendo de las Historias de Usuario realizadas anteriormente se construirá una planificación de distintas Iteraciones basándose en el tiempo y procurando agrupar la funcionalidad realizada en la misma iteración. De esta manera continuar con el avance del proyecto proporcionando especificaciones para su desarrollo e implementación.

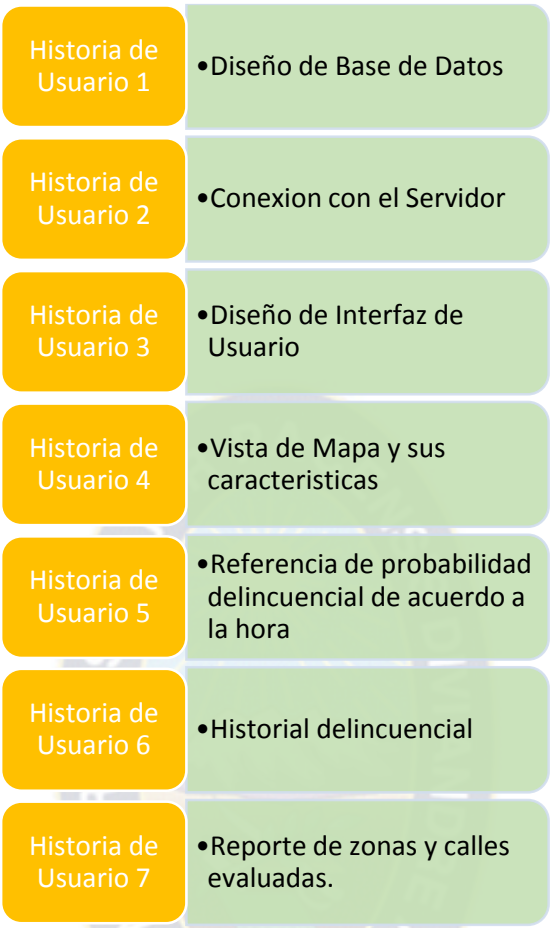


Figura 3.2: Diagrama de Historia de Usuario
Fuente: Elaboración propia

3.4. Fase de Producción y Estabilización

El propósito de esta fase es implementar la funcionalidad requerida en el producto mediante la aplicación iterativa y el ciclo de desarrollo incremental.

3.4.1. Diseño del software como servicio

A continuación se presentará un esquema de lo que será el software como servicio FELLSAFE con relacion a la sincronizacion en la nube y su extraccion de datos.

Por definición, el propósito de la fase de estabilización es la integración de subsistemas más pequeños desarrollados por diferentes equipos en un solo producto. Las actividades que se realizaron durante esta fase son exactamente lo mismo con las actividades llevadas a cabo durante los días de trabajo y por lo tanto los artefactos equipos generalmente crean son semánticamente iguales que los artefactos que hemos creado en las fases anteriores. Por lo tanto, los resultados de esta tarea se finalicen de arquitectura, diseño y documentos de interfaz de usuario.

Figura 3.3: Diseño de software como servicio y Mobile Service
Fuente: Elaboración propia

3.4.2. Diseño del sistema con Webml

Webml, propone un diseño de datos, compatible con el modelo entidad relación o con el modelo de clases UML, debido a que utiliza una nomenclatura y definición similar (Entidad, Atributo, Relaciones)

3.4.2.1. Modelo de Datos Estructural

A continuación se muestra el Modelo de Datos estructural realizada con WebRatio detallando las entidades que se presentaran en el software y sus determinados atributos.

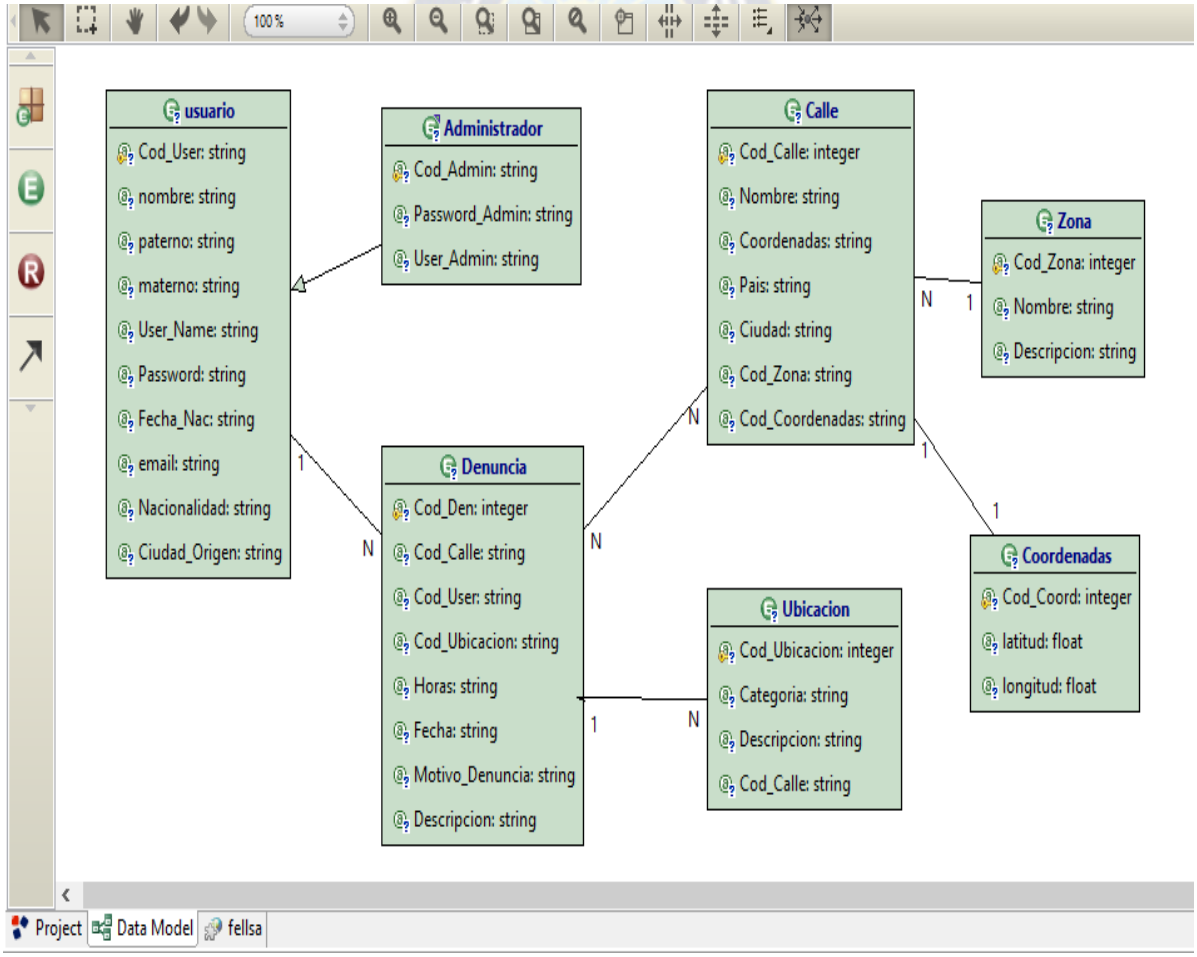


Figura 3.4: Modelo de Datos estructural FEELSAFE
Fuente: Elaboración propia

3.4.2.2. Modelo de Hipertexto

Modelo de Hipertexto Administrador

Main Page

Página cuyo contenido será visible durante el ingreso a la aplicación durante la primera vista.

Como contenido de usuario devuelve sus datos al haber accedido a la aplicación, enlazada con un Multi Message Unit que nos permite mostrar al usuario un mensaje.

En este mensaje va el nombre del usuario verificado así como un enlace para cerrar la sesión del mismo.

Módulo de gestión de Usuarios

Módulo con todo lo necesario para poder realizar la gestión de los usuarios. Con este módulo podremos realizar las operaciones típicas (Crear, Leer, Modificar, Leer) sobre la entidad.

Administrar de forma correcta el software y sus respectivos usuarios.

Módulo de gestión de denuncias

Módulo con todo lo necesario para poder realizar la gestión de los módulos de la aplicación.

Con este módulo podremos realizar las de salida de informes sobre las denuncias de la cual deriva a distintas interfaces.

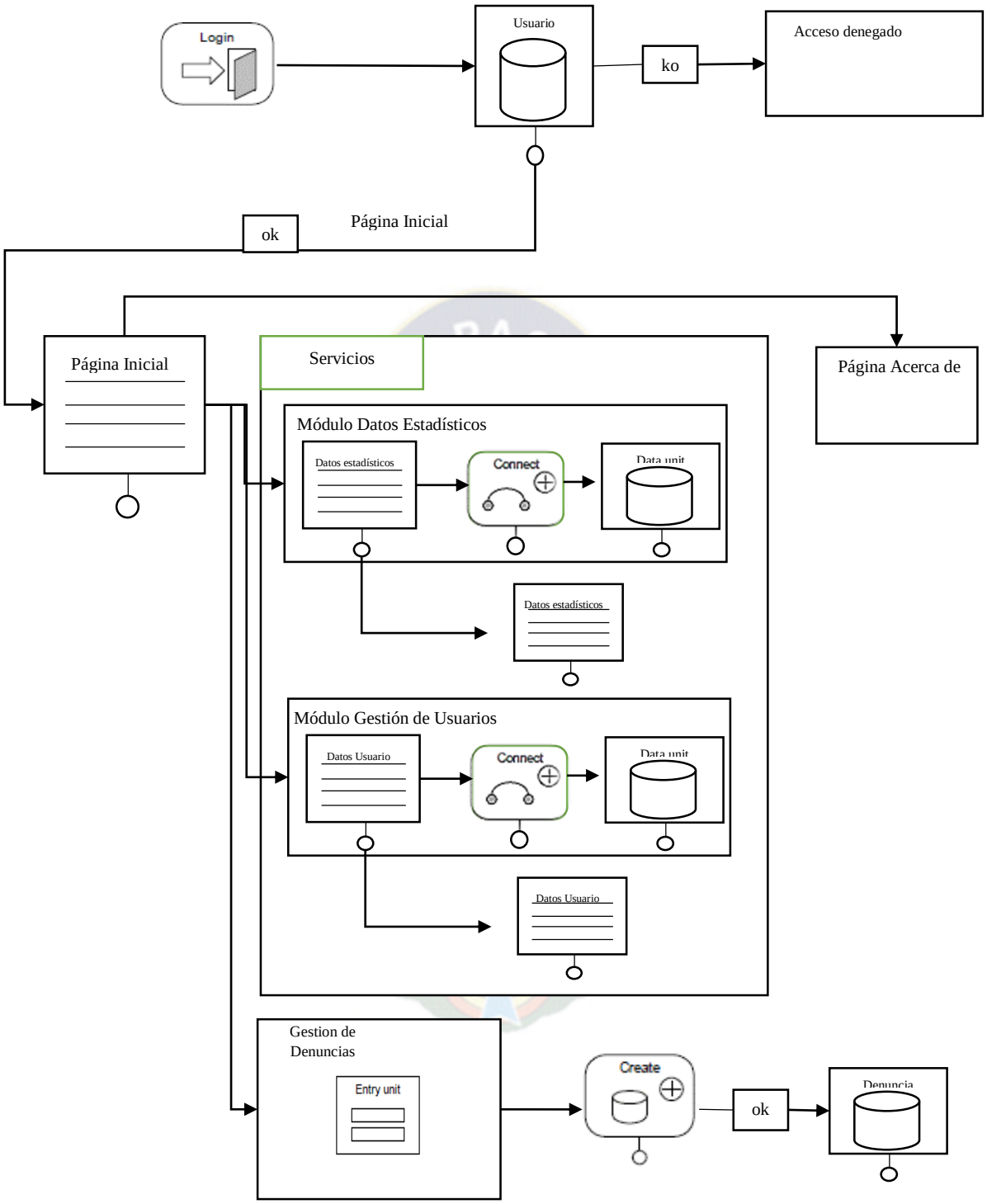


Figura 3.5: Modelo de Hipertexto Administrador
Fuente: Elaboración propia

3.4.2.3. Descripción de la Interfaz de Usuario

La interfaz de usuario es el medio en el cual el usuario puede comunicarse con el software a través de un dispositivo móvil, de los cuales su entorno debe ser amigable y fácil de usar.

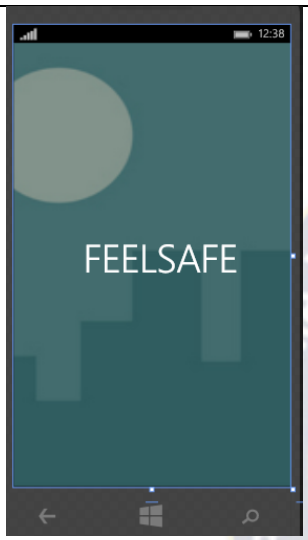
PANTALLA DE BIENVENIDA	
	Es la Pantalla donde Se muestra el Nombre de la Aplicación, junto a un diseño de bienvenida al usuario

Tabla 3.11.: Pantalla de Bienvenida
Fuente: Elaboración Propia

A continuación se muestra la pantalla de acceso a la aplicación FEELSAFE para el usuario.

ACCESO A LA APLICACIÓN	
	En esta pantalla se encuentra un formulario de acceso a la aplicación y un botón de registro esto para los usuarios nuevos al servicio móvil.

Tabla 3.12.: Acceso a la Aplicación
Fuente: Elaboración Propia

Se muestra la vista de mapa y sus atributos en la aplicación, detallando sus características y funciones.

DATOS GEORREFERENCIALES	
	Se muestra una pantalla en la que el usuario se podrá encontrar a través del GPS en el mapa mostrando su ubicación y las coordenadas en la que está.

Tabla 3.13.: Pantalla de Datos georreferenciales
Fuente: Elaboración Propia

A continuación se muestra la pantalla de reportes delincuenciales en la aplicación.

SEGMENTACION DEL MAPA POR ZONA	
	En esta pantalla se muestra el mapa utilizado en diferentes colores de referencia, esto debido a la intensidad de criminalidad relacionada con la Zona o calle.

Tabla 3.14.: Pantalla Segmentación del mapa por zona
Fuente: Elaboración Propia

Por consiguiente tenemos la vista de denuncia de usuario, esto para beneficio del mismo, en su uso personal.

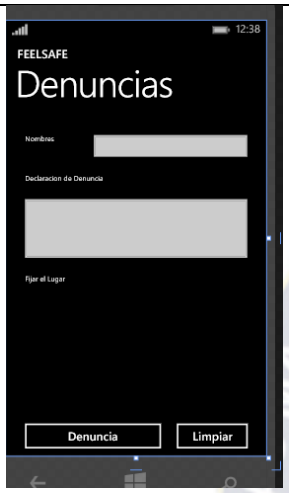
DENUNCIA DE USUARO	
	Pantalla en la que un usuario puede denunciar el lugar donde encuentre actos delictivos, y puede detallar el tipo de agavio o incidente que sufrio o vio.

Tabla 3.15.: Pantalla de Denuncia de Usuario
Fuente: Elaboración Propia

Esta vista es el reporte de usuarios en el sistema la cual está en una versión beta ya que solo seria una vista para el administrador.

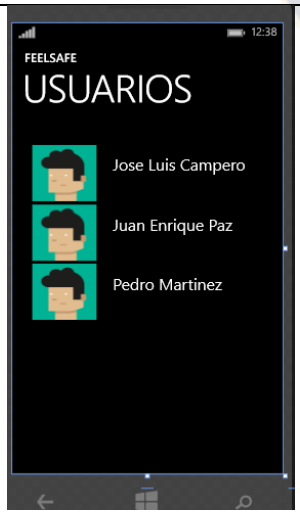
LISTADO DE USUARIOS	
	En esta pantalla se muestran los usuarios que utilizan la aplicación. Algunos con aportes para la misma.

Tabla 3.16.: Pantalla de listado de Usuarios
Fuente: Elaboración Propia

A continuación se muestra una lista de lugares riesgosos de los cuales se puede detallar el contenido.

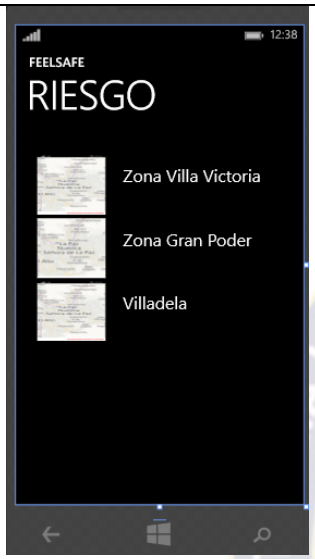
LISTA DE LUGARES	
	En esta pantalla se muestran los lugares que han tenido al menos un acto delictivo y mostrando sus detalles y el nivel de criminalidad que existe en su radio.

Tabla 3.17.: Pantalla de Lista de Lugares
Fuente: Elaboración Propia

3.4.3. Desarrollo del software como servicio

3.4.3.1 Iteraciones

a) Iteración 1 Diseño de la Base de Datos

- **Historia de usuario**

Numero/ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Dedicado		
H1	Nuevo	Moderado	Difícil	20 hrs	20 hrs	Alta	
Descripción							
El diseño de la Base de datos debe estar debidamente enfocada a los datos que el administrador y usuario le proporcionan. En la aplicación el usuario debe tener acceso a la información solicitada. Se debe tener rápido y fácil acceso a la información de lugares con alto índice de criminalidad							
Fecha	**Estado**	**Comentarios**					
15.08.15	Definición	Se ha determinado la Base de datos de manera manual detallando las entidades y relación entre ellas					

15.08.15	Implementación	La base de datos ha sido implementada en la Nube.
12.10.15	Finalización	La aplicación ya tiene acceso a la base de datos
12.10.15	Verificado	Se mejoraron los cambios necesarios en la funcionalidad del servicio móvil.

Tabla 3.18.: Historia Diseño de la Base de Datos
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

Numero/ID	**Tipo**	**Dificultad**		**Confidencialidad**	**Esfuerzo**		**Notas**
		Antes	Después	1(no confidencial) 4(muy confidencial)	Estimado	Actual	
Buscar Tarea 1	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **3**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	2	20 hrs.	20 hrs.	
Descripción							
Las Bases de Datos Normalizadas y creadas a partir de un gestor de bases de Datos en la Nube.							
Fecha	**Estado**	**Comentarios**					
15.08.15	Definición	La base de datos implementada debe utilizarse.					
15.08.15	Implementación	El análisis mostro que hay tablas que describen a las entidades del software.					
12.10.15	Finalización	Las pruebas unitarias a la funcionalidad a la base de datos se comunican de manera síncrona al servicio móvil.					
12.10.15	Verificado	Se ejecutaron las pruebas de manera exitosa.					

Tabla 3.19.: Tarjeta de Tarea Diseño de la Base de Datos
Fuente: [Elaboración Propia]

- **Test Aceptación**

Aceptación Hoja de prueba
Test ID: 1
Historia: Tarea 1
Fecha Escrita: Quince de Agosto de Dos mil Quince
Fecha de Ejecución: 12.10.15
Aprobado / Rechazado ID: Aprobado
Descripción:

- La base de datos se habilito correctamente para el uso del servicio móvil. - Implementación de la base de datos en la Nube.
Resultado Esperado - Implementación de la base de datos - El servicio móvil está conectado a la base de datos en la Nube.

Tabla 3.20.: Hoja de Prueba de Diseño de la base de datos
Fuente: Elaboración Propia

- **Modelo de Hipertexto Altas bajas y cambios de la Base de Datos**

A continuación se muestran las altas bajas y cambios en general de las entidades de la base de datos.

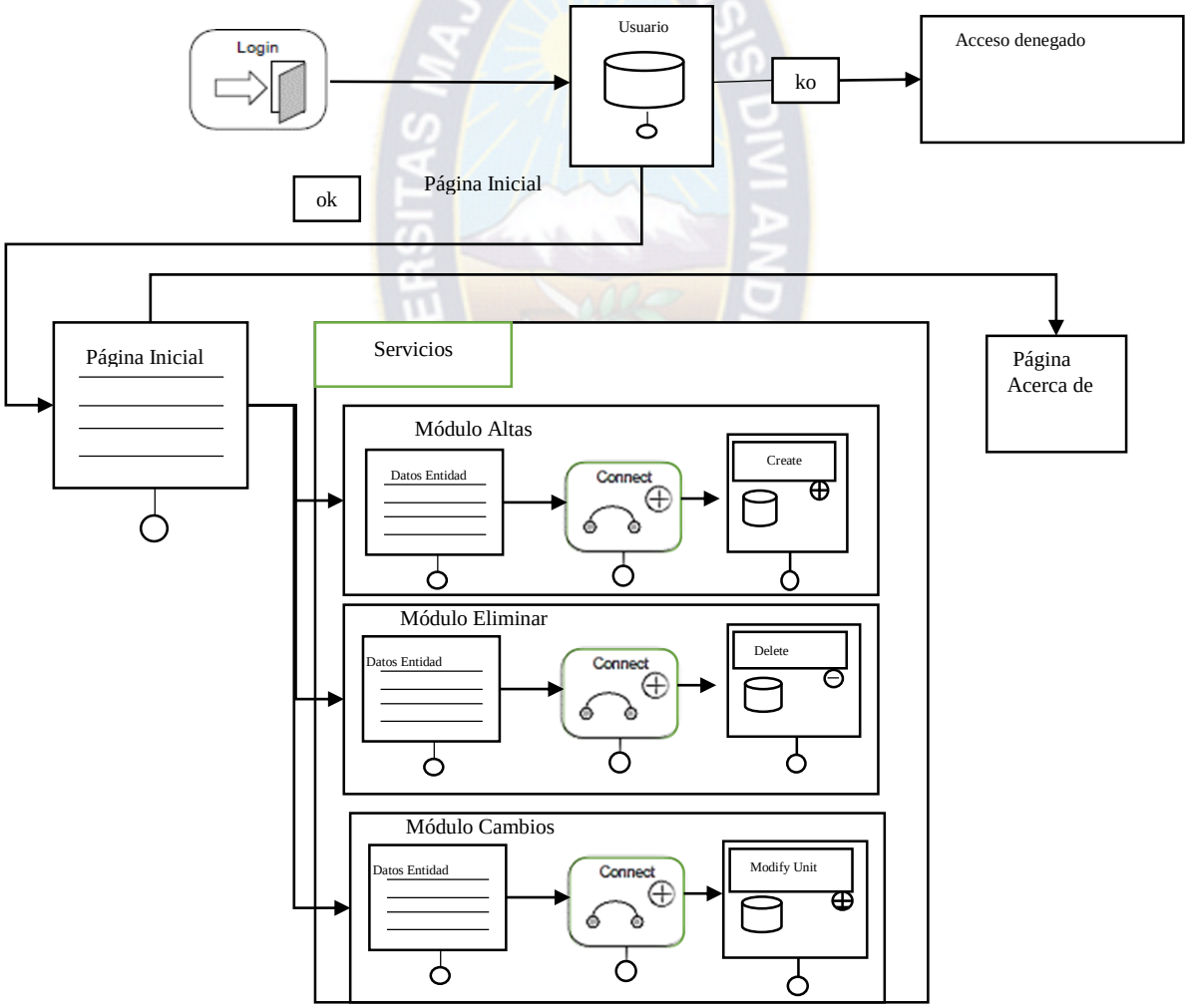


Figura 3.6: Modelo de Hipertexto Modelado de la Base de datos
Fuente: [Elaboración propia]

b) Iteración 2 Conexión con el servidor

- **Historia de usuario**

Numero/ID	**Tipo**	**Dificultad**		**Esfuerzo**		**Prioridad**	**Notas**
		Antes	**Después Numero**	**Estimado**	**Dedicado**		
H2	Nuevo	Moderado	Difícil	20 hrs	20 hrs	Alta	
Descripción							
Establecer la conexión con Microsoft Azure para el uso de la base de datos en la Nube.							
Fecha	**Estado**	**Comentarios**					
25.08.15	Definición	Se realiza la conexión al servidor de manera automática al momento de iniciar la aplicación.					
25.08.15	Implementación	La Conexión con la base de datos ha sido implementada en la Nube.					
18.10.15	Finalización	El servicio móvil ya tiene acceso a la aplicación móvil.					
18.10.15	Verificado	La conexión al servidor ha sido realizada exitosamente					

Tabla 3.21.: Historia Conexión con el Servidor
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

/ID	**Tipo**	**Dificultad**		**Confidencialidad**	**Esfuerzo**		**Notas**
		Antes	**Después**	**1(no confidencial) 4(muy confidencial)**	Estimado	Actual	
Buscar Tarea 2	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **4**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	**4**	15 hrs	20 hrs	
Descripción							
La conexión a la nube con Microsoft Azure, posibilita que el servicio móvil obtenga acceso a la base de datos de manera segura.							
Fecha	**Estado**	**Comentarios**					
25.08.15	Definición	La conexión con el servidor es utilizada al momento de iniciar la aplicación móvil utilizando el servicio web.					
25.08.15	Implementación	El análisis mostro que la conexión con el servidor es de suma importancia por el acceso a la información contenida en la base de datos					

18.10.15	Finalización	La aplicación móvil esta sincronizada con el servicio móvil, eso quiere decir que la conexión con la base de datos esta implementada
18.10.15	Verificado	Las pruebas con la conexión se ejecutan de manera exitosa.

Tabla 3.22.: Tarjeta de Tarea Conexión con el servidor
Fuente: [Elaboración Propia]

- **Test de Aceptación**

Aceptación Hoja de prueba
Test ID: 2
Historia: Tarea 2
Fecha Escrita: Veinticinco de Agosto de Dos mil Quince
Fecha de Ejecución: 18.10.15
Aprobado / Rechazado ID: Aprobado
Descripción: - El servidor se encuentra habilitado para el uso correcto del servicio móvil - Implementación con la conexión al servidor en la Nube.
Resultado Esperado - Conexión de la aplicación móvil con el servicio móvil. - El servidor y sus recursos están conectados al servicio móvil.

Tabla 3.23.: Hoja de Prueba de Conexión con el servidor
Fuente: Elaboración Propia

- **Modelo de Hipertexto Altas bajas y cambios de la Base de Datos**

A continuación se muestran la conexión con el servidor

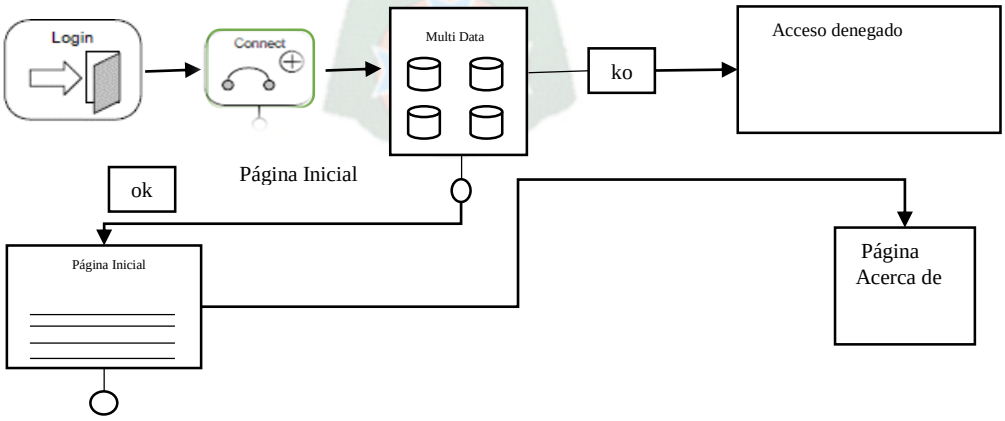


Figura 3.7: Modelo de Hipertexto Conexión con el Servidor
Fuente: [Elaboración propia]

c) **Iteración 3.- Diseño de la Interfaz de Usuario**

- **Historia de usuario**

Numero/ID	**Tipo**	**Dificultad**		**Esfuerzo**		**Prioridad**	**Notas**
		Antes	**Después**	**Estimado**	**Dedicado**		
H3	Nuevo	Moderado	Moderado	15 hrs	20 hrs	7	
Descripción							
Diseñar un entorno gráfico amigable de fácil manejo para los usuarios.							
Fecha	**Estado**	**Comentarios**					
19.09.15	Definición	La aplicación móvil debe tener una interfaz utilizada por el usuario					
19.09.15	Implementación	La interfaz está a disposición del usuario					
10.10.15	Finalización	Las pruebas de aceptación usuario/Interfaz han sido realizadas					
10.10.15	Verificado	Las pruebas finalizaron y se ejecutaron exitosamente.					

Tabla 3.24.: Historia Diseño de la Interfaz de Usuario
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

Numero/ ID	**Tipo**	**Dificultad**		**Confidencialidad**	**Esfuerzo**		**Notas**
		Antes	**Después**	**1(no confidencial) 4(muy confidencial)**	Estimado	Actual	
Buscar Tarea 3	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **4**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	**3**	15 hrs	20 hrs	
Descripción							
La interfaz de una aplicación es de suma importancia para los usuarios, de hecho mientras mejor sea la interfaz, mas atraído se sentirá el usuario.							
Fecha	**Estado**	**Comentarios**					
10.09.15	Definición	La interfaz de usuario debe ser de uso cotidiano para el usuario					
10.09.15	Implementación	De acuerdo al análisis debe facilitarse la interfaz para el usuario.					
10.10.15	Finalización	Las interfaces de distintos módulos han sido creadas.					

10.10.15	Verificado	Las pruebas se realizaron con una ejecución exitosa.

Tabla 3.25.: Tarjeta de Tarea Diseño de la Interfaz de Usuario
Fuente: [Elaboración Propia]

- **Test de Aceptación**

Aceptación Hoja de prueba
Test ID: 3
Historia: Tarea 3
Fecha Escrita: Diecinueve de Septiembre de Dos mil Quince
Fecha de Ejecución: 10.10.15
Aprobado / Rechazado ID: Aprobado
Descripción: - La interfaz de usuario ya está lista para enlazar los datos - Implementación de la interfaz de usuario con modelo vista vista modelo
Resultado Esperado - Interfaz amigable para el usuario - El servicio móvil tiene interfaz de usuario.

Tabla 3.26.: Hoja de Prueba de Diseño de la Interfaz de Usuario
Fuente: Elaboración Propia

- **Modelo de Hipertexto Diseño de la Interfaz de usuario**

A continuación se muestra el Hipertexto de Interfaz de usuario.

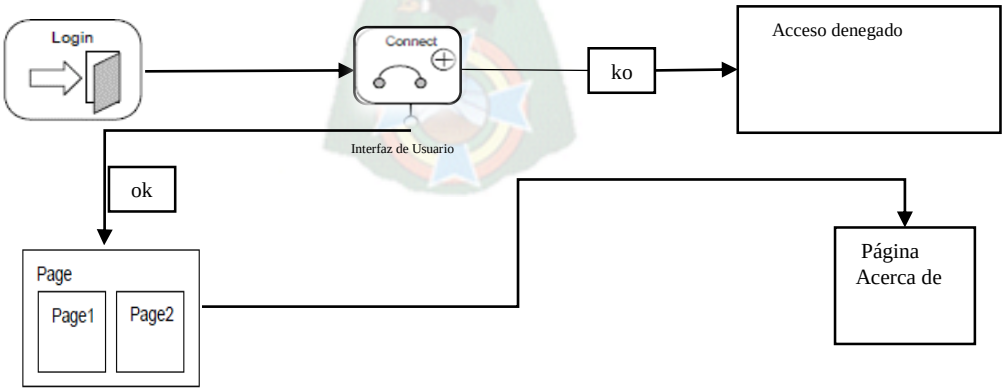


Figura 3.8: Modelo de Hipertexto Interfaz de Usuario
Fuente: [Elaboración propia]

d) Iteración 4.- Vista de Mapa y sus características

- **Historia de usuario**

Numero/ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Dedicado		
H4	Nuevo	Moderado	Moderado	15 hrs	20 hrs	Alta	
Descripción:							
Se debe tener al acceso al mapa donde se encuentre el usuario mediante su ubicación por GPS. Diseñar una interfaz gráfica que coincida con el mapa y referencie puntos establecidos.							

Fecha	Estado	Comentarios
11.10.15	Definición	En la vista del mapa debe visualizarse la posición del usuario. Y las características del software.
11.10.15	Implementación	Se establece el mapa en la aplicación móvil.
09.11.15	Finalización	Los cambios necesarios de la funcionalidad en el servicio móvil han sido implementados en el mapa.
09.11.15	Verificado	Se utilizó el uso de mapa mejorando su uso para el usuario

Tabla 3.27.: Historia Vista de Mapa y sus características
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

Numero/ ID	Tipo	Dificultad		Confidencialidad	Esfuerzo		Notas
		Antes	Después	1(no confidencial) 4(muy confidencial)	Estimado	Actual	

Buscar Tarea 4	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **3**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	4	15 hrs	20 hrs	
Descripción							
Se deberá cumplir con la ingeniería de requerimientos para denotar que atributos tiene el mapa y los datos almacenados a través de su uso.							
Fecha	**Estado**	**Comentarios**					
11.10.15	Definición	De acuerdo a lo establecido se debe plantear el uso del mapa para distintos escenarios.					
11.10.15	Implementación	El análisis mostró que el uso del mapa y el uso de sus atributos son necesarios.					
19.11.15	Finalización	La inclusión del mapa en la aplicación móvil ha sido establecida.					
19.11.15	Verificado	Las pruebas se realizaron con una ejecución exitosa.					

Tabla 3.28.: Tarjeta de Tarea Vista de Mapa y sus características
Fuente: [Elaboración Propia]

- **Test de Aceptación**

Aceptación Hoja de prueba
Test ID: 4
Historia: Tarea 4
Fecha Escrita: Once de octubre de Dos mil Quince
Fecha de Ejecución: 19.11.15
Aprobado / Rechazado ID: Aprobado
Descripción: - La vista de Mapa contiene todos los requisitos que se pide - Implementación de la vista mapa para el manejo del servicio móvil.
Resultado Esperado - Vista de mapa interactivo - El servicio móvil tiene interfaz de uso de mapa.

Tabla 3.29: Hoja de Prueba vista de mapa y sus características.
Fuente: Elaboración Propia

- **Modelo de Hipertexto Diseño de la vista de mapa y sus características**

A continuación se muestra el Hipertexto de vista de mapa y sus características.

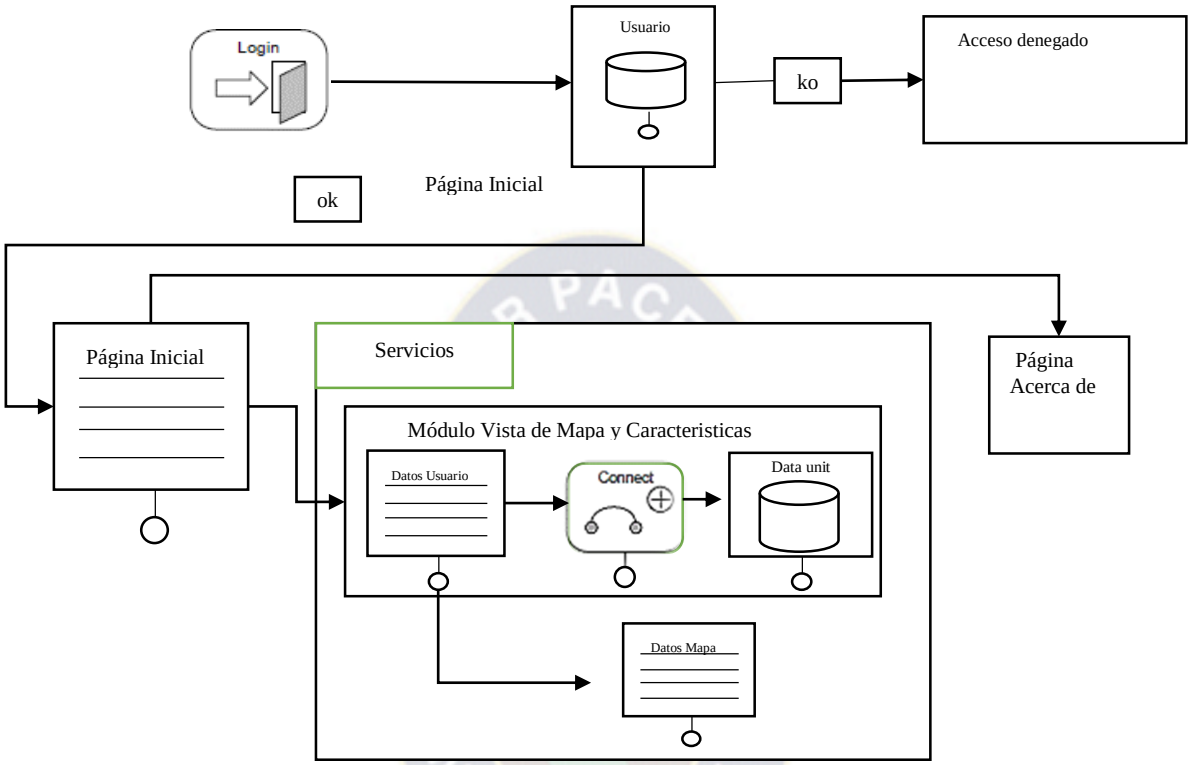


Figura 3.9: Modelo de Hipertexto vista de mapa y sus características
Fuente: [Elaboración propia]

e) Iteración 5.- Referencia de probabilidad delincuencial de acuerdo a la hora

- **Historia de usuario**

Numero/ID	**Tipo**	**Dificultad**		**Esfuerzo**		**Prioridad**	**Notas**
		Antes	**Después**	**Estimado**	**Dedicado**		
H5	Nuevo	Moderado	Moderado	15 hrs	20 hrs	**Alta**	
Descripción							
Los datos referenciados en la aplicación tendrán referencia a determinada hora relacionando los reportes con la hora actual en la que este el usuario.							
Fecha	**Estado**	**Comentarios**					
05.10.15	Definición	Establecer dentro de la interfaz un temporizador y a partir de este generar los reportes en la aplicación.					

15.10.15	Implementación	La implementación del módulo Reporte de Estados
22.11.15	Finalización	Llenado de información referencial a hechos delictivos en determinada hora dentro de la base de datos en la nube.
22.11.15	Verificado	Todas las pruebas se realizaron con éxito.

Tabla 3.30.: Historia referencia de probabilidad delincuencial de acuerdo a la hora
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

Numero/ ID	**Tipo**	**Dificultad**		**Confidencialidad**	**Esfuerzo**		**Notas**
		Antes	**Después**	**1(no confidencial) 4(muy confidencial)**	Estimado	Actual	
Buscar Tarea 5	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **3**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	4	15 hrs	20 hrs	
Descripción							
Establecer dentro de la interfaz datos referenciales a la hora y un reloj, a partir de este generar los reportes en la aplicación.							
Fecha	**Estado**			**Comentarios**			
05.10.15	Definición			Llenado de Información referencial a hechos delictivos en determinada hora dentro de la base de datos en la Nube.			
15.10.15	Implementación			El análisis mostro que la implementación de alertas en determinada hora es una innovación			
22.11.15	Finalización			La referencia de probabilidad delincuencial ha sido implementada en la interfaz.			
22.11.15	Verificado			Las pruebas se realizaron con una ejecución exitosa.			

Tabla 3.31.: Tarjeta de Tarea referencia de probabilidad delincuencial de acuerdo a la hora
Fuente: [Elaboración Propia]

- **Test de Aceptación**

Aceptación Hoja de prueba
Test ID: 5
Historia: Tarea 5
Fecha Escrita: Quince de Octubre Dos mil Quince
Fecha de Ejecución: 22.11.15

Aprobado / Rechazado ID: Aprobado
Descripción: - Datos de acumulación de datos delincuenciales en determinada hora - Implementación del módulo Referencia de probabilidad delincuencial de acuerdo a la hora ha sido implementada.
Resultado Esperado - Vista del Módulo referencia de probabilidad delincuencial de acuerdo a la hora, para el uso de su información en la interfaz. - La aplicación tiene un reporte detallado para determinados lugares.

Tabla 3.32: Hoja de Prueba de referencia de probabilidad delincuencial de acuerdo a la hora
Fuente: [Elaboración Propia]

f) Iteración 6.- Historial delincuencial

- **Historia de usuario**

Numero/ID	**Tipo**	**Dificultad**		**Esfuerzo**		**Prioridad**	**Notas**
		Antes	**Después**	**Estimado**	**Dedicado**		
H6	Nuevo	Moderado	Moderado	15 hrs	20 hrs	Alta	
Descripción							
De acuerdo a un tiempo establecido se actualizarán los niveles estadísticos de los reportes sobre criminalidad con ayuda de los usuarios, en la aplicación.							

Fecha	**Estado**	**Comentarios**
30.10.15	Definición	Realizar un módulo de denuncia para el usuario referenciando lugares y descripción del hecho ocurrido.
30.10.15	Implementación	El historial delincuencial se implementara realizando consultas a la base de datos y proporcionando las características necesarias
24.11.15	Finalización	La interfaz de usuario tiene acceso al Historial delincuencial de zonas.
24.11.15	Verificado	Todas las pruebas se realizaron con éxito.

Tabla 3.33.: Historia Historial delincuencial
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

Numero/ ID	Tipo	Dificultad		Confidencialidad	Esfuerzo		Notas
		Antes	**Después**	**1(no confidencial) 4(muy confidencial)**	Estimado	Actual	
Buscar Tarea 6	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **3**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	4	15 hrs	20 hrs	
Descripción							
Establecer dentro de la interfaz formularios declarativos para el usuario y determinar lugares y hechos delictivos con ayuda del formulario.							
Fecha	**Estado**			**Comentarios**			
30.10.15	Definición			Llenado de Información referencial a hechos delictivos y su descripción.			
30.10.15	Implementación			El análisis de las denuncias por el usuario determinó la robustez de la información si esta es debidamente proporcionada.			
24.11.15	Finalización			El historial delincuencial ha sido implementado en la interfaz.			
24.11.15	Verificado			Las pruebas se realizaron con una ejecución exitosa.			

Tabla 3.34.: Tarjeta de Tarea Historial delincuencial
Fuente: [Elaboración Propia]

- **Test de Aceptación**

Aceptación Hoja de prueba
Test ID: 6
Historia: Tarea 6
Fecha Escrita: Treinta de Octubre Dos mil Quince
Fecha de Ejecución: 24.11.15
Aprobado / Rechazado ID: Aprobado
Descripción: - Acumulación de datos por parte del usuario al software a través de formularios facilitados.

- Implementación del módulo Referencia de probabilidad delincuencial de acuerdo a la hora ha sido implementada.
Resultado Esperado - Vista del Módulo Historial delincuencial, para el uso de su información en la interfaz. - La aplicación contiene un módulo de interacción con el usuario para que pueda proporcionar datos de denuncia.

Tabla 3.35: Hoja de Prueba de Historial delincuencial
Fuente: Elaboración Propia

- **Modelo de Hipertexto Historial delincuencial**

A continuación se muestra el Hipertexto de Historial delincuencial.

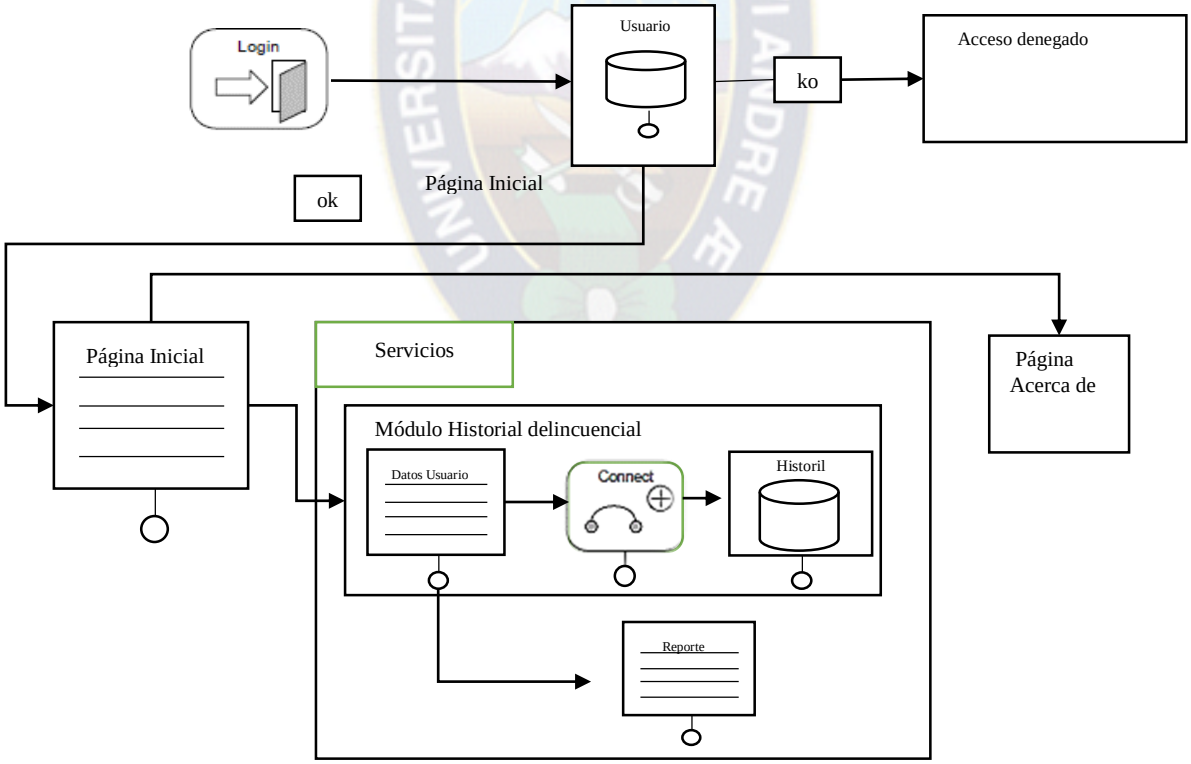


Figura 3.10: Modelo de Hipertexto Historial delincuencial
Fuente: [Elaboración propia]

g) Iteración 7.- Reporte de zonas y calles evaluadas.

- **Historia de usuario**

Numero/ID	Tipo	Dificultad		Esfuerzo		Prioridad	Notas
		Antes	Después	Estimado	Dedicado		
H7	Nuevo	Moderado	Moderado	15 hrs	20 hrs	Alta	
Descripción							
Los datos contenidos en la base de datos se evaluaran para mostrar en la aplicación un rango de colores respecto a la criminalidad.							
Fecha	**Estado**			**Comentarios**			
02.11.15	Definición			Módulo de reportes obtenidos de la base de datos y realizados por denuncias de los usuarios.			
02.11.15	Implementación			El Reporte de zonas y calles evaluadas se implementara realizando consultas a la base de datos y extrayendo las características necesarias			
27.11.15	Finalización			La aplicación ya tiene acceso a los reportes facilitados a través del servicio móvil.			
27.11.15	Verificado			Se mejoraron los cambios necesarios en funcionalidad del servicio móvil.			

Tabla 3.36.: Historia Reporte de zonas y calles evaluadas.
Fuente: [Elaboración Propia]

- **Tarjeta de Tarea**

Numero/ID	Tipo	Dificultad		Confidencialidad	Esfuerzo		
		Antes	Después	1(no confidencial) 4(muy confidencial)	Estimado	Actual	
Buscar Tarea 7	Nuevo	Tiempo de ejecución (1) – muy dificultoso (5) **3**	Tiempo de ejecución (1) – muy dificultoso (5) **4**	4	15 hrs	20 hrs	

Descripción		
De acuerdo a un color establecido de peligro, el usuario identificara mejor los lugares de riesgo en determinada zona y observara su rango de criminalidad.		
Fecha	**Estado**	**Comentarios**
02.11.15	Definición	Se utilizan marcadores para detallar distintos puntos de alto peligro en el mapa y un texto generado.
02.11.15	Implementación	Se implementó uso de marcadores dentro del mapa de la aplicación móvil.
27.11.15	Finalización	Se demostró que el uso de marcadores para detallar zonas y calles evaluadas es óptimo para la aplicación.
27.11.15	Verificado	Las pruebas se realizaron con ejecución exitosa.

Tabla 3.37.: Tarjeta de Tarea Reporte de zonas y calles evaluadas.
Fuente: [Elaboración Propia]

- **Test de Aceptación**

Aceptación Hoja de prueba
Test ID: 7
Historia: Tarea 7
Fecha Escrita: Dos de Noviembre Dos mil Quince
Fecha de Ejecución: 27.11.15
Aprobado / Rechazado ID: Aprobado
Descripción: - Reporte de zonas y calles evaluadas muestran los requisitos que se piden. - Implementación del módulo Reporte de zonas y calles evaluadas para el manejo del servicio móvil. -
Resultado Esperado - Vista del Módulo Reporte de zonas y calles evaluadas para el uso del usuario. - El servicio móvil tiene interfaz de Reporte de zonas y calles evaluadas.

Tabla 3.38: Hoja de Prueba de Reporte de zonas y calles evaluadas.
Fuente: [Elaboración Propia]

- **Modelo de Hipertexto Reporte de zonas y calles evaluadas**

A continuación se muestra el Hipertexto de Reporte de zonas y calles evaluadas.

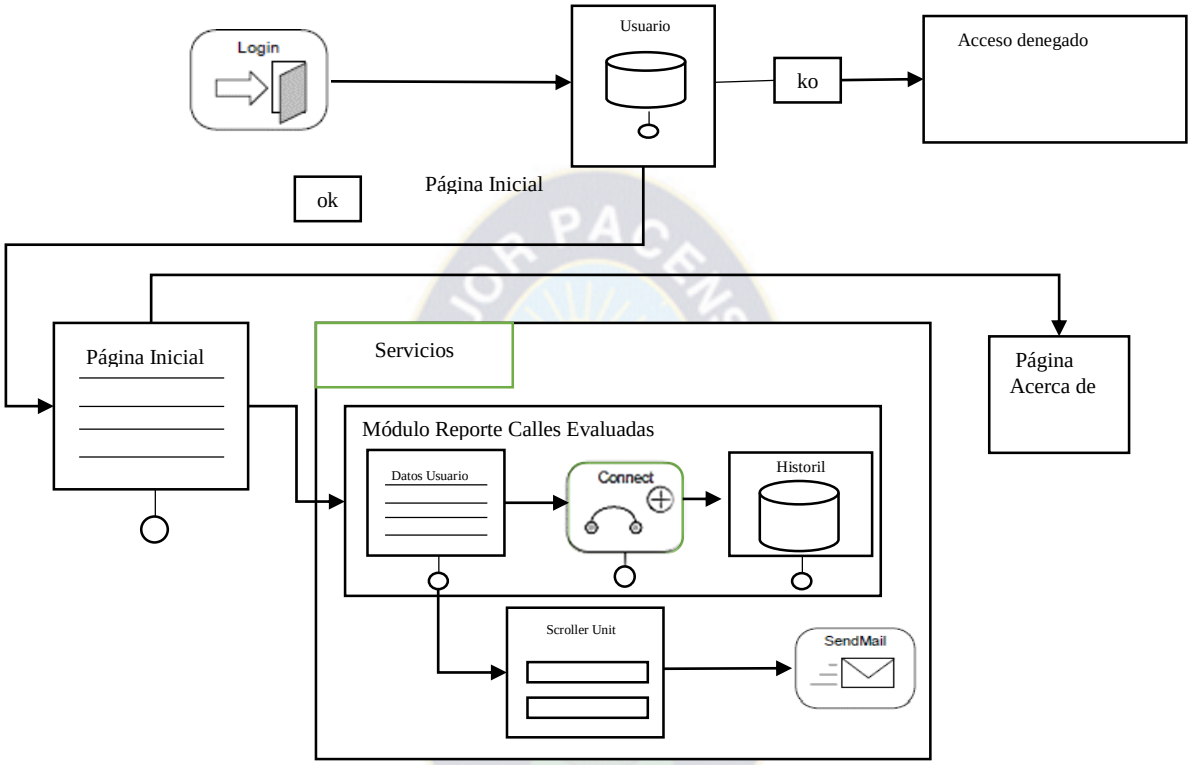


Figura 3.11: Modelo de Hipertexto Reporte de zonas y calles evaluadas.
Fuente: [Elaboración propia]

3.5. Fase de Pruebas del Sistema

Con esta fase concluimos la metodología Mobile – D describiendo brevemente las pruebas se han realizado al servicio móvil.

Algunos factores se toman en cuenta al momento de la evaluación, para las pruebas del servicio móvil, se determina la funcionalidad de cada módulo utilizando pruebas de usuario, así pretendiendo llegar al objetivo de la investigación.

Un factor que debe cumplir el servicio móvil es la usabilidad, que se aplica con la adaptación del usuario y la aplicación móvil.

3.5.1. Prueba de Dispositivos.

Para realizar las pruebas de se contempló utilizar dispositivos Nokia Lumia de distinta resolución y categoría, debido a que la aplicación está desarrollada especialmente para este tipo de dispositivos.

- Se logra determinar el factor de ubicación en el mapa, con un adecuado funcionamiento del receptor GPS de cada dispositivo.
- Respecto a cada módulo se determinó el tiempo de espera para cargar la nueva información obtenida.
- Se realiza el estudio de resolución de pantallas en cada dispositivo.

3.5.2. Pruebas de Uso

Un factor que debe cumplir este servicio móvil es la usabilidad, que se refiere al interés que el usuario le dé a esta aplicación móvil y de forma paralela su distribución por motivo de descarga en la tienda de aplicaciones. La resultante de este estudio ha sido detallada en la prueba de hipótesis en el siguiente capítulo.

- Conocimiento de aplicaciones móviles que sirvan para la prevención delincuencial.
- Interés por obtener una aplicación que informe sobre lugares peligrosos en una determinada zona.
- Usabilidad de dicha aplicación y frecuencia de uso.
- Conocimiento de métodos preventivos con la ayuda de un dispositivo móvil.

CAPÍTULO VI

ANALISIS DE DATOS Y RESULTADOS

4.1. Prueba de Hipótesis

En esta sección realizaremos la evaluación de la hipótesis planteada en el capítulo primero.

<<El uso de Cloud Computing, Programación móvil, sobre una plataforma de servicio en la nube, facilita la información de mapas referido a la georreferenciación y geolocalización de manera remota e independiente, los cuales se utilizarán para dar soluciones a problemas de criminalidad y dar paso a la prevención delincuencial en la ciudad de La Paz>>

De este proceso identificamos las variables, independiente, dependiente, e interviniente.

- Variable Independiente: georreferenciación, geolocalización.
- Variable dependiente: Prevención delincuencial, problemas de criminalidad.
- Variable interviniente: Plataforma de servicio en la Nube, Cloud Computing.

4.1.1. Contraste de Rachas de Wald - Wolfowitz

La prueba de rachas (también llamada prueba de Wald - Wolfowitz después de Abraham Wald y Jacob Wolfowitz) es una no paramétrica prueba estadística que comprueba la hipótesis de aleatoriedad de una secuencia de datos de dos valores.

Supongamos una población cuya función de distribución es desconocida y sea X la variable aleatoria asociada a esa población, la cual solo puede tomar dos posibles valores como ejemplo, éxito (A) y fracaso (B):

H_0: La muestra aleatoria

H_1: La muestra no es aleatoria

En general, sea una muestra de tamaño *n* en la que han aparecido n_1 elementos de tipo A y n_2 elementos del tipo B, siendo $n_1 + n_2 = n$ sea la variable aleatoria:

R: Número total de rachas en la muestra

Para muestra grande y bajo la hipótesis H_0 es decir, para muestras aleatorias la distribución de probabilidad de R tiende hacia la normal a medida que n_1 y n_2 se van haciendo grandes. Esta aproximación es bastante buena si $n_1 > 10$ y $n_2 > 10$; de tal manera que:

$\boldsymbol{R \rightarrow N(E\,[R], \sqrt{Var}[R])}$	
Esperanza	$E[R] = \dfrac{2n_1n_2}{n_1 + n_2}$
Varianza	Var[R] = $\dfrac{2n_1n_2(2n_2n_2 - n_1 - n_2)}{(n_1+n_2)^2\,(n_1+n_2-1)}$

Tabla 4.1: Esperanza y Varianza
Fuente: [Elaboración Propia]

Por consiguiente para muestras grandes se verifica:

$$Z = \frac{R - E[R]}{\sqrt{Var}[R]}$$

Y para una muestra concreta el calor de estadístico Z será:

$$Z_{exp} = \frac{R - (\frac{2n_1n_2}{n} + 1)}{\sqrt{\frac{2n_1n_2(2n_1n_2 - n)}{n^2(n-1)}}} + 1$$

En donde R es el número total de rachas observadas en la muestra.

La región de aceptación para la hipótesis nula será:

$$-Z_{\frac{\alpha}{2}} < Z_{exp} < Z_{\frac{\alpha}{2}}$$

El valor de $Z_{\frac{\alpha}{2}}$ se obtiene de la tabla de la N (0.1) de manera que:

$$P\left(Z_1 \leq -Z_{\frac{\alpha}{2}}\right) = p\left(Z_1 \geq Z_{\frac{\alpha}{2}}\right) = \frac{\alpha}{2}$$

Desarrollo de la Prueba de Hipotesis

Para el desarrollo de la prueba de hipótesis por medio de contraste de rachas de wald-wolfowitz se sigue los siguientes pasos:

Paso 1. Planteamiento de la Hipótesis nula.

H_i: Prevención delincuencial, problemas de criminalidad.

Paso 2. Selección el nivel de confianza.

El nivel de confianza o significancia que se elige es $\alpha = 0.05$ elegida de la Tabla Normal.

Paso 3. Identificación de estadístico de prueba

Para este caso se utiliza la prueba de rachas de Wald-Wolfowitz utiliza os signos de los residuos y sus variaciones de negativos y positivos o viceversa. Una racha vendrá constituida por la sucesión de signos iguales.

Paso 4. Formulación de la regla de decisión.

Para la prueba, nos planteamos una encuesta de valoración del servicio móvil para la prevención de criminalidad urbana, la cual estudia los usuarios que necesitan de una aplicación móvil de esta categoría, y su aceptación sobre el software como servicio.

De los resultados de la encuesta, tenemos la siguiente tabla:

Nro.	Preguntas	Software como servicio	Aceptación por rachas
1.	Estructura General del servicio móvil	Mejora	+
2.	Acceso a la aplicación móvil.	Mejora	+
3.	Introducción datos usuario	Mejora	+
4.	Ubicación del usuario en el mapa a través de geolocalización.	Mejora	+
5.	Registro de denuncias por parte de los usuarios frente a algún atentado delincuencial.	Indiferente	-
6.	Verificación actualizada de acuerdo al número de veces en las que el usuario se conecte a Internet.	Mejora	+
7.	Visualización datos de lugar observado.	Mejora	+
8.	Visualización de Historial delincuencial	Mejora	+
9.	Visualización del Mapa y sus características.	Mejora	+
10.	Visualización de datos delincuenciales por determinada hora.	Mejora	+
11.	Visualización de datos de usuario	Indiferente	-

Tabla 4.2. Propuesta de software como servicio y rango de aceptación de usuario
Fuente: [Elaboración propia]

Se obtienen los siguientes resultados:

(+ + + +)(-)(+ + + + +)(-)

Donde:

- (-) Representa los casos en los que el usuario muestra desinterés o un interés mínimo.
- (+) Representa casos en los que el servicio móvil mostro interés por parte del usuario en comparación a otros medios.
- Siendo una racha construida por la sucesión de signos iguales se tiene:

Total de Rachas expuestas	$R_{exp} = 4$
Número total de observaciones	$N = 11$
Número de residuos positivos	$n_1 = 9$
Número de residuos negativos	$n_2 = 2$

- Reemplazando daos para calcular la Esperanza y Varianza se tiene:

Esperanza	$E[R] = \frac{2n_1n_2}{n_1+n_2} + 1 = \frac{2*9*2}{9+2} + 1 = \frac{36}{11} + 1 = 4.27$
Varianza	Var[R] $= \frac{2n_1n_2(2n_2n_2-n_1-n_2)}{(n_1+n_2)^2\,(n_1+n_2-1)} = \frac{2*9*2(2*9*2-9-2)}{(9+2)^2(9+2-1)}$ $= \frac{900}{1210} = 0.7438$

Paso 5: Toma de decisión

Y para una muestra concreta el valor del estadístico Z_{exp} reemplazando datos se tiene:

- $Z_{exp} = \frac{R-E[R]}{\sqrt{Var[R]}} + 1 = \frac{4-4.27}{\sqrt{0.7438}} + 1 = \frac{-0.27}{0.86324} + 1 = 0.6872$

- Para calcular la región de aceptación de la hipótesis es necesario halar el valor de $Z\alpha_{/2}$ que se obtiene de la tabla de la N (0,1), de manera que cumple:

$$P\left(Z_1 \leq -Z_{\frac{\alpha}{2}}\right) = p\left(Z_1 \geq Z_{\frac{\alpha}{2}}\right) = \frac{\alpha}{2}$$

$P\left(Z_1 \leq -Z_{\frac{\alpha}{2}}\right) = \frac{\alpha}{2}$	$P\left(Z_1 \geq Z_{\frac{\alpha}{2}}\right) = \frac{\alpha}{2}$
$1 - P\left(Z_1 < Z_{\frac{\alpha}{2}}\right) = \frac{\alpha}{2}$	$\left(Z_1 \geq Z_{\frac{\alpha}{2}}\right) = 0.025$
$P\left(Z_1 < Z_{\frac{\alpha}{2}}\right) = 1 - \frac{\alpha}{2}$	$Z\alpha_{/2} = 1.96$
$P\left(Z_1 < Z_{\frac{\alpha}{2}}\right) = 1 - 0.025$	
$P\left(Z_1 < Z_{\frac{\alpha}{2}}\right) = 0.975$	
$Z\alpha_{/2} = 1.96$	

- Por tanto la región de aceptación para la hipótesis nula es:
  - $-Z\alpha_{/2} < Z_{exp} < Z\alpha_{/2}$
  - **-1.96 < 0.6872 < 1.96**
- Se puede ver que el estadístico Z_{exp} = 0.6872 se encuentra en el intervalo de la hipótesis, por lo tanto se puede afirmar que H_i: Prevención delincuencial, problemas de criminalidad.
- Lo que demuestra que la tesis es un trabajo válido, además muestra que los datos de la muestra son aleatorios.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Luego del diseño, implementación, y prueba de forma preliminar el Software como Servicio para la prevención de criminalidad urbana utilizando métodos de georreferenciación, en este capítulo presentamos los datos obtenidos durante la investigación.

- La implementación del software como servicio para la prevención de criminalidad urbana utilizando métodos de georreferenciación, requiere la interacción con el usuario de manera frecuente para su mejor uso y filtración de datos
- El acceso instantáneo desde cualquier punto, presenta mayor confiabilidad al momento de saber por qué lugares ir, y que lugares del mapa evitar.
- De los resultados obtenidos se pretende que la implementación de este software sea una ayuda para la ciudadanía, como método de prevención contra abusos delincuenciales.
- El servicio móvil (Mobile service) optimiza los tiempos de peligro para diferentes ubicaciones en el mapa.
- Se logró controlar la duplicidad de registros a través de una base de datos implementada en la nube.
- La plataforma de Microsoft en la nube permite la mayor seguridad de datos para el servicio móvil (Mobile Service).
- El objetivo general fue cumplido, que es realizar un software como servicio para la prevención de criminalidad urbana utilizando métodos de georreferenciación.
- El uso de servicios en la nube es algo innovador, por lo tanto se logró implementar un servicio utilizado por una aplicación móvil.
- La factibilidad que da la plataforma de Microsoft en la Nube ha sido muy beneficiosa para el desarrollo de esta investigación.

- De acuerdo al resultado obtenido de la hipótesis, se concluye que el “Software como servicio para la prevención de criminalidad urbana utilizando métodos de georreferenciación” es factible.

5.2. Recomendaciones

Concluyendo el proyecto de investigación de tesis se recomiendan los siguientes puntos tanto para uso del software como en desarrollo del mismo:

- Se recomienda tener en cuenta nuevas herramientas de uso cotidiano para la prevención de criminalidad debido a que existe en muchos lugares del país, y estos son inseguros para las personas que no están informadas de los modos de operación delincuencial.
- La presente investigación se limita al uso de la plataforma de Microsoft, por lo que es necesario volverlo multiplataforma y así tener un uso conjunto para la sociedad.
- Es conveniente que esta idea sea utilizada en algunas entidades de seguridad, de este modo poder hacer un estudio más detallado sobre lugares de alto peligro para la sociedad.
- Es recomendable que esta aplicación se utilice no solo en el país si no internacionalmente.
- Se recomienda a los usuarios la participación en lo que es la retroalimentación sobre críticas de la aplicación y de este modo mejorarla para versiones futuras.
- Esta aplicación puede ser usada para el ámbito turístico, comprobando que personas que llegan del exterior o el interior desean saber que lugares son de su conveniencia.

6. Bibliografía

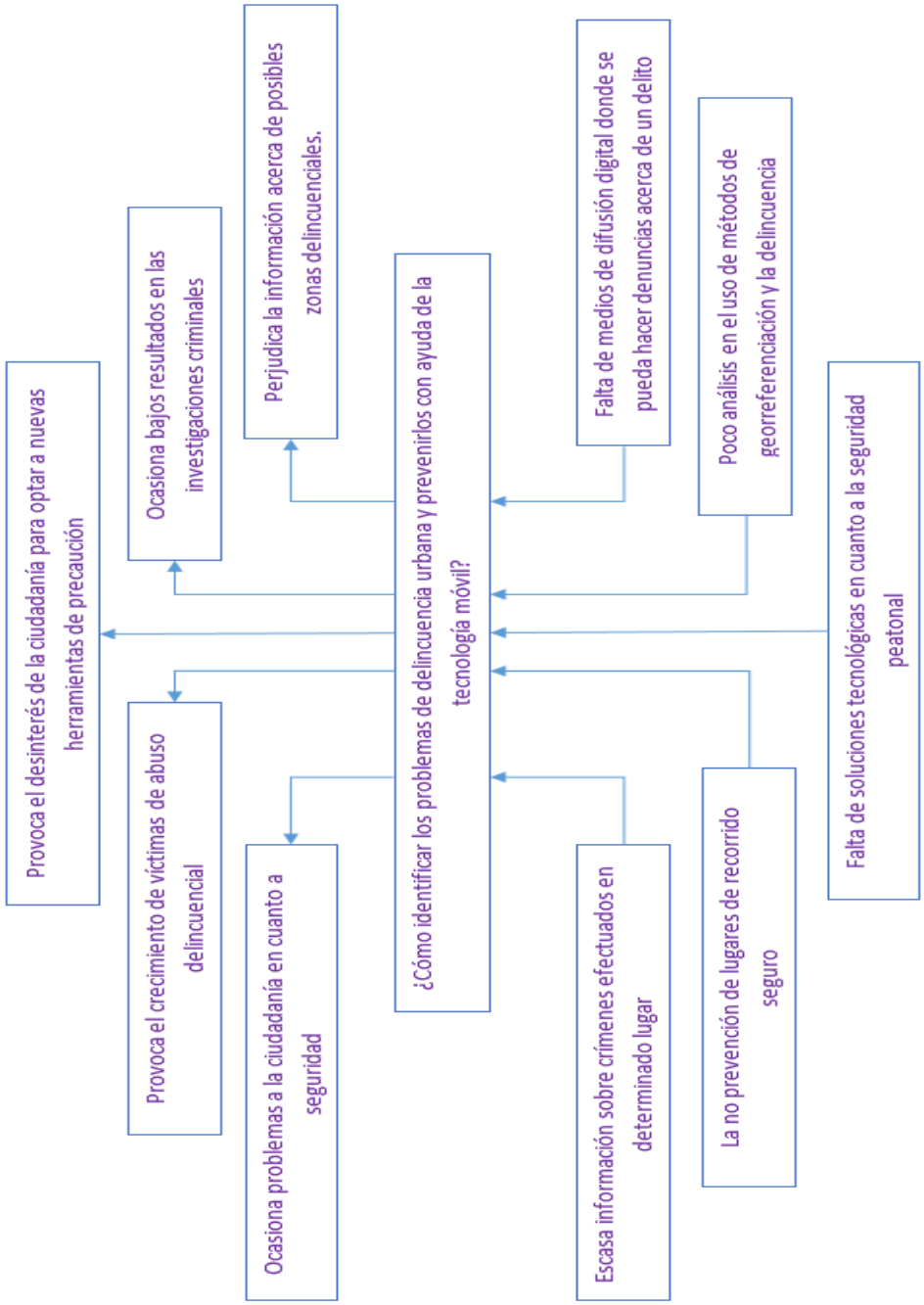
1. BING MAPS – MICROSOFT. THE BING MAPS PLATFORM A MAPPING SOLUTION BUILT FOR THE ENTERPRISE. Obtenido en: https://www.microsoft.com/maps/
2. BLANCO P, CAMARERO J, FUMERO A, WERTERSKI A, y RODRIGUEZ P. 2009. METODOLOGIA DE DESARROLLO AGIL PARA SISTEMAS MOVILES INTRODUCCION AL DESARROLLO CON ANDROID. Obtenido en: <http://www.adamwesterski.com/wpcontent/files/docsCursos/Agile_doc_TemasAnv.pdf> [consulta: Junio del 2015].
3. BRAMBILLA M, COMAI S, FRATERNALI P, MATERA M. DESIGNING WEB APPLICATIONS WITH WEBML AND WEBRATIO [s.a.]. Obtenido en: <http://www.webml.org/webml/upload/ent5/1/Chapter%209%20-%20WebML.pdf > [Consulta: mayo, 2015].
4. CAMPS C, y ORIOL A. 2012. LA NUBE: OPORTUNIDADES Y RETOS PARA LOS INTEGRANTES DE LA CADENA DE VALOR Obtenido en: <http://www.managementsolutions.com/PDF/ESP/La-nube.pdf> [Consulta: mayo del 2015].
5. CUPI J. M. 2014. GEORREFERENCIACION Y SU USO EN SMARTPHONES PARA LA LOCALIZACION DE PERSONAS [Consulta: Junio del 2015].
6. CONDORI L (Condori, 2014). 2014. SISTEMA DE INFORMACIÓN GERENCIAL PARA EL CONTROL Y SEGUIMIENTO DE LAS ESPECIES SILVESTRES EN ÁREAS URBANAS: CASO LA PAZ [Consulta: Junio del 2015].
7. LANDA I. 2011. WINDOWS AZURE Obtenido en: <https://es.scribd.com/doc/72993459/Subete-a-la-nube-de-Microsoft> [Consulta: Junio del 2015]
8. MARQUEZ A. 2008. HISTORIA DE LOS SISTEMAS OPERATIVOS. Obtenido en: <http://sistemasdistribuido.blogspot.com/2008/08/2-historia-de-los-sistemas-operativos.html> [consulta: Junio del 2015]

9. MENDOZA O. 2011. CLOUD - COMPUTING. Obtenido en: <http://www.monografias.com/trabajos-pdf5/cloud-computing/cloud-computing.shtml> [consulta: Junio del 2015]

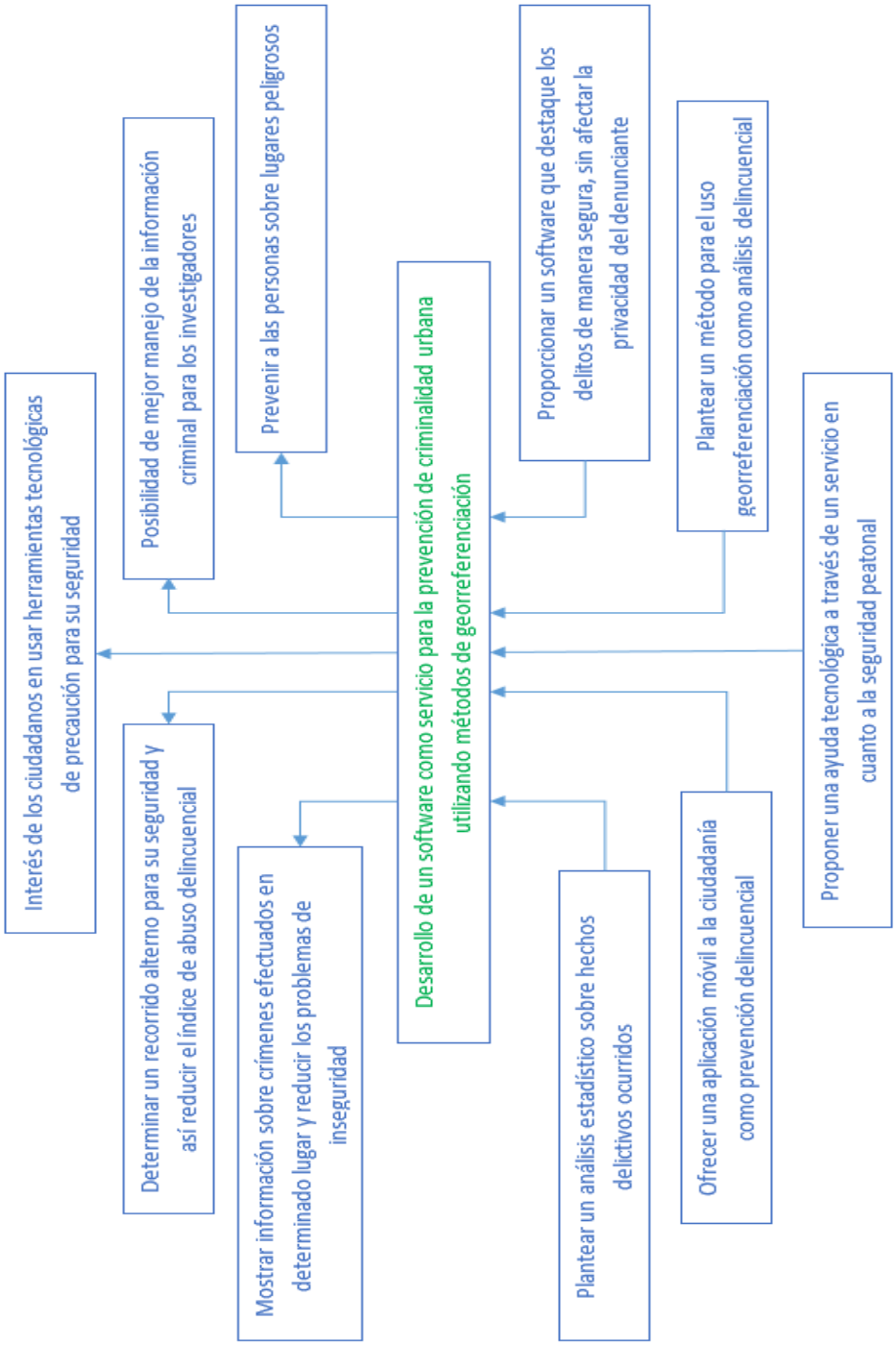
10. JOSKOWICZ J. 2008. REGLAS Y PRACTICAS EN EXTREMEPROGRAMMING Obtenido en: <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf> [Consulta: mayo Del 2015].
11. HERNANDEZ R, FERNANDEZ C, y PILAR L. 2006. METODOLOGIA DE LA INVESTIGACION
Obtenido en: <https://competenciashg.files.wordpress.com/2012/10/sampieri-et-al-metodologia-de-la-investigacion-4ta-edicion-sampieri-2006_ocr.pdf>
[Consulta: mayo, 2015].
12. SCHENONE M. 2004. DISEÑO DE UNA METODOLOGÍA ÁGIL DE DESARROLLO DE SOFTWARE.
Obtenido en: <http://materias.fi.uba.ar/7500/schenone-tesisdegradoingenieriainformatica.pdf> [Consulta: mayo, 2015].
13. TECNOLOGICO DE MONTERREY 2010. EJEMPLO DE MATRIZ DE MARCO LOGICO
Obtenido en: <http://www.cca.org.mx/funcionarios/cursos/sam/html/m5/pdf/ejemplo-matriz.pdf > [Consulta: mayo del 2015].
14. YAHUITA Y. 2014. SOFTWARE COMO SERVICIO PARA LA ADMINISTRACIÓN DE HISTORIAS CLÍNICAS [Consulta: Junio del 2015].

ANEXOS

Anexo A – Árbol de Problemas



Anexo B – Árbol de objetivos



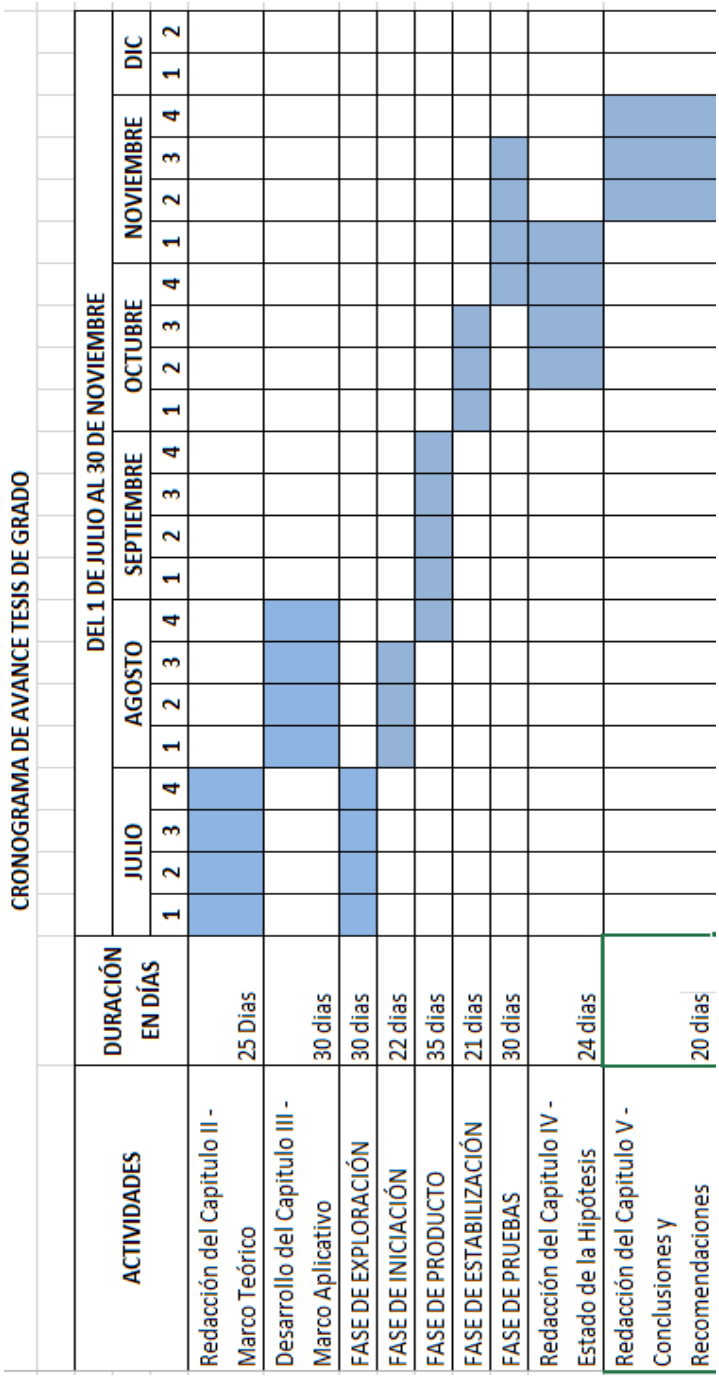
Anexo C – Matriz de Marco Lógico

RESUMEN NARRATIVO DE LOS OBJETIVOS	INDICADORES	FUENTES DE VERIFICACIÓN	SUPUESTOS
FIN Lo que se espera alcanzar al desarrollar este proyecto es reducir los índices de criminalidad en la ciudadanía. Por ahora La Paz – Bolivia y en un futuro a nivel internacional.	De acuerdo al diagrama de Gantt proporcionado anteriormente se destaca una tentativa conjunto al cronograma del desarrollo para el proyecto.	Los planes de trabajo se realizaran de acuerdo a un análisis inicial sobre las estadísticas obtenidas del gobierno municipal y la policía nacional.	Los usuarios continuaran usando la aplicación propuesta para la prevención delincuencial.
PROPÓSITO Desarrollo de un software como servicio para la prevención de criminalidad urbana utilizando métodos de georreferenciación.	El desarrollo del proyecto se realizara acorde a las fechas ya planteadas como tentativa.	Estadísticas del departamento de investigación anticrimen de la FELCC Estadísticas del Gobierno Municipal de La Paz	Reducción considerable de la criminalidad y delincuencia. Los usuarios retroalimentan constantemente la información del software.
RESULTADOS - Plantear un análisis estadístico sobre hechos delictivos ocurridos, que muestren información sobre crímenes	- Estudio analítico sobre hechos delictivos que muestren los lugares detectados. Y disminución de	- Registros de evaluación en el transcurso del proyecto sobre la información obtenida.	- Resultados que apoyen el estudio de manera sustancial para su buen uso.

efectuados en determinado lugar y reduzcan los problemas de inseguridad.	problemas de seguridad		
- Ofrecer una aplicación móvil a la ciudadanía como prevención delincuencial que determine un recorrido alterno para su seguridad y así reducir el índice de abuso delincuencial.	- Disminución del abuso delincuencial gracias a la proporción de una aplicación móvil.	- Registros de lugares peligrosos y estudio de la reducción o incremento con la criminalidad y la delincuencia.	- Constante participación de los usuarios para la clarificación de lugares peligrosos.
- Proponer una ayuda tecnológica a través de un servicio móvil cuanto a la seguridad peatonal que aliente el interés a los ciudadanos en usar herramientas tecnológicas de precaución para su seguridad.	- Incremento del interés ciudadano para el uso de herramientas tecnológicas como prevención delincuencial.	- Registro de descargas según el tiempo transcurrido después de la publicación de dicha aplicación.	- Nuevas propuestas de los usuarios al desarrollador para la mejora de la aplicación.
- Plantear un método para el uso georreferenciación como análisis delincuencial, que posibilite el mejor manejo de la información criminal para los investigadores.	- Disminución delincuencial con ayuda de métodos de georreferenciación para el estudio de zonas de alto criminal para la ciudadanía.	- Registro visual sobre zonas peligrosas según un color respectivo al cálculo de peligro en estas.	- Verificación exacta en determinados cuadrantes del mapa proporcionado.
- Proporcionar un software en el cual se pueda destacar los delitos de manera segura y sin afectar la privacidad del denunciante, con esto	- Porcentaje de personas asaltadas reducida gracias al	- Registro de delitos en determinadas zonas y su modus operandi.	- Mejoramiento constante del software, mostrando

prevenir a las personas sobre lugares peligrosos.	software de prevención.	- Registro de usuarios con ayuda de redes sociales como Facebook.	información de entendimiento sencillo a los usuarios.
ACTIVIDADES Primero se realizaran estudios sobre la criminalidad de los cuales estarán separados por dos secciones: - Se estudiara la criminalidad conjunto a la georreferenciación para verificar cuales son los lugares de alto índice criminal y delincuencial. - Se estudiara cuáles son los motivos de la criminalidad y su crecimiento según datos estadísticos proporcionados por la Alcaldía y la FELCC.	- Desarrollo de una aplicación móvil que utilice métodos de georreferenciación y geolocalización - Desarrollo de software como servicio para estudio y prevención delincuencial en la ciudadanía. - Estudio de frecuencia de personas asaltadas en determinado lugar de acuerdo a un análisis estadístico.	- Estadísticas de uso de la aplicación. - Estadísticas de prevención ciudadana respecto a la delincuencia y criminalidad. - Estadísticas y registros sobre descargas de la aplicación proporcionada.	Se cuenta con recursos para la elaboración y desarrollo de dicho proyecto. Existe disposición de las entidades para proporcionar información verídica sobre la delincuencia.

Anexo D cronograma de avance



Anexo E

Encuesta propuesta y sus resultantes

P1. ¿Cuenta usted con un celular Inteligente “Smartphone”?

- Si
- No

P2. ¿Conoce usted algún método digital de prevención delincuencial en su país y ciudad?

- Sí, tengo conocimiento
- No, Eso no existe en mi ciudad o país.

P3. ¿Cuántos tipos de aplicaciones móviles conoce usted para la prevención de criminalidad?

- 0
- 1 – 5
- Mayor a 5

P4. ¿Cree usted que en el país existen métodos de prevención delincuencial con ayuda de aplicaciones móviles?

- Si
- No

P5. ¿Cree usted que es necesaria una aplicación móvil para el control de criminalidad urbana?

- Muy Necesario
- Necesario
- Innecesario
- No es importante

P6. ¿Qué pensaría usted si existiese una aplicación móvil que le sirva para prevenir algún tipo de delincuencia a su persona?

- Es muy útil.
- No es muy útil

P7. ¿En el momento de ir a algún lugar desconocido para su persona, utilizaría usted alguna aplicación móvil que le ayude a saber el grado de peligro?

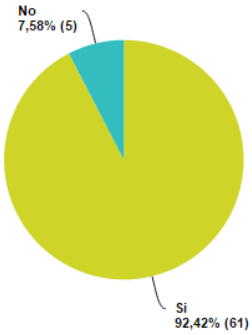
- Frecuentemente.
- Algunas veces
- Nunca

P8 ¿Conoce usted alguna aplicación (web, móvil online, móvil offline, etc.) en donde pueda saber que lugares de su ciudad son peligrosos?

- Si
- No

¿Cuenta usted con un celular Inteligente "Smart Phone"?

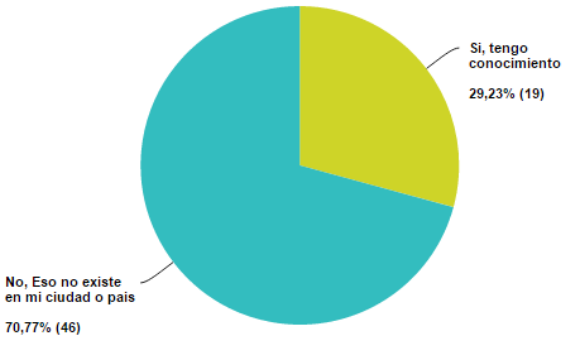
Respondido: 66 Omitido: 1



Opciones de respuesta	Respuestas	
Si	92,42%	61
No	7,58%	5
Total		66

¿Conoce usted algún método digital de prevención delincuencial en su país y ciudad?

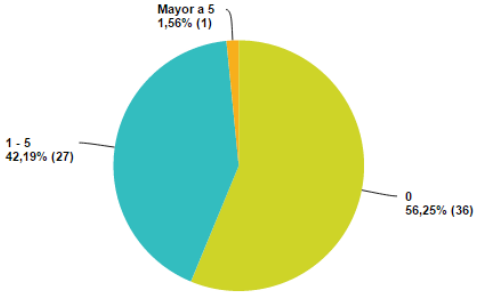
Respondido: 65 Omitido: 2



Opciones de respuesta	Respuestas	
Si, tengo conocimiento	29,23%	19
No, Eso no existe en mi ciudad o pais	70,77%	46
Total		65

¿Cuántos tipos de aplicaciones móviles conoce usted para la prevención de criminalidad?

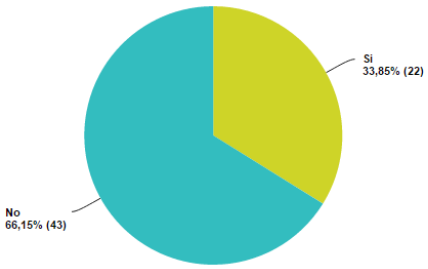
Respondido: 64 Omitido: 3



Opciones de respuesta	Respuestas	
0	56,25%	36
1 - 5	42,19%	27
Mayor a 5	1,56%	1
Total		64

¿Cree usted que en el país existen métodos de prevención delincuencial con ayuda de aplicaciones móviles?

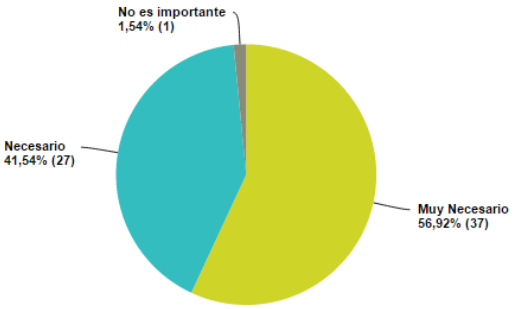
Respondido: 65 Omitido: 2



Opciones de respuesta	Respuestas	
Si	33,85%	22
No	66,15%	43
Total		65

¿Cree usted que es necesaria una aplicación móvil para el control de criminalidad urbana?

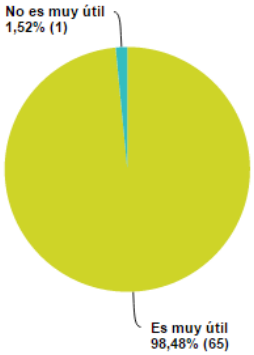
Respondido: 65 Omitido: 2



Opciones de respuesta	Respuestas	
Muy Necesario	56,92%	37
Necesario	41,54%	27
Innecesario	0,00%	0
No es importante	1,54%	1
Total		65

¿Qué pensaría usted si existiese una aplicación móvil que le sirva para prevenir algún tipo de delincuencia a su persona?

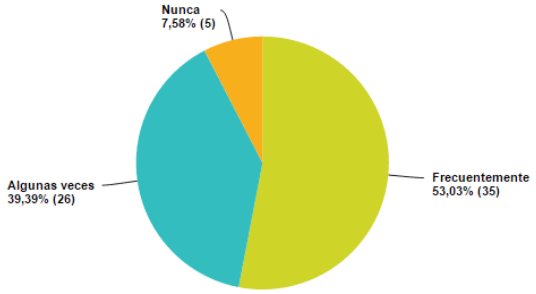
Respondido: 66 Omitido: 1



Opciones de respuesta	Respuestas	
Es muy útil	98,48%	65
No es muy útil	1,52%	1
Total		66

¿En el momento de ir a algún lugar desconocido para su persona, utilizaría usted alguna aplicación móvil que le ayude a saber el grado de peligro?

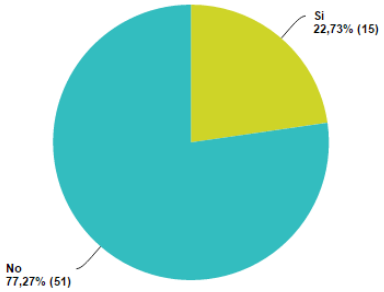
Respondido: 66 Omitido: 1



Opciones de respuesta	Respuestas	
Frecuentemente	53,03%	35
Algunas veces	39,39%	26
Nunca	7,58%	5
Total		66

¿Conoce usted alguna aplicación (web, móvil online, móvil offline, etc.) en donde pueda saber que lugares de su ciudad son peligrosos?

Respondido: 66 Omitido: 1



Opciones de respuesta	Respuestas	
Si	22,73%	15
No	77,27%	51
Total		66