



UNIVERSIDAD TECNOLÓGICA EQUINOCCIAL

**FACULTAD DE CIENCIAS DE LA INGENIERÍA
INGENIERÍA INFORMÁTICA Y CIENCIAS DE LA
COMPUTACIÓN**

**APLICATIVO DE RASTREO PARA SISTEMAS ANDROID,
MEDIANTE UNA PLATAFORMA WEB Y MENSAJES DE
TEXTO EN CASO DE PERDIDA O ROBO.**

**TRABAJO PREVIO A LA OBTENCIÓN DEL TÍTULO
DE INGENIERO EN INFORMÁTICA Y CIENCIAS DE LA COMPUTACIÓN**

SANTIAGO MAURICIO BORJA VITERI

DIRECTOR: Dr. Diego Ordoñez

Quito, mayo 2015

© Universidad Tecnológica Equinoccial. 2015

Reservados todos los derechos de reproducción

DECLARACIÓN

Yo **SANTIAGO MAURICIO BORJA VITERI**, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Tecnológica Equinoccial puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

Santiago Mauricio Borja Viteri

C.I. 1719212571

CERTIFICACIÓN

Certifico que el presente trabajo que lleva por título “**Aplicativo de rastreo para sistemas Android, mediante una plataforma web y mensajes de texto en caso de pérdida o robo**”, que, para aspirar al título de Ingeniero Informática y Ciencias de la computación fue desarrollado por **Santiago Borja**, bajo mi dirección y supervisión, en la Facultad de Ciencias de la Ingeniería; y cumple con las condiciones requeridas por el reglamento de Trabajos de Titulación artículos 18 y 25.

Dr. Diego Ordóñez
DIRECTOR DEL TRABAJO
C.I. 1710449016

DEDICATORIA

El presente trabajo de grado dedico a mi familia, especialmente a mi madre Martha Viteri y a mi padre quienes me motivaron e impulsaron a esforzarme cada día más.

Santiago Borja

AGRADECIMIENTO

Agradezco a la Universidad Tecnológica Equinoccial, institución que me dio la oportunidad de formarme como profesional.

A todos mis profesores que dejaron enseñanzas en mí, durante el trayecto recorrido.

A mi director de tesis por haberme brindado los lineamientos para poder culminar el presente trabajo y llegar a la meta final.

Santiago Borja

ÍNDICE DE CONTENIDOS

RESUMEN	xiii
ABSTRACT	xiv
INTRODUCCIÓN	1
MARCO TEÓRICO	4
2.1. APLICACIÓN MÓVIL	4
2.2. SISTEMAS WEB	4
2.3. SMARTPHONE	5
2.4. SERVICIOS GOOGLE	5
2.4.1. GOOGLE CLOUD MESSAGING (GCM)	6
2.5. ANDROID	6
2.5.1. ACTIVITY	6
2.5.2. SERVICE	7
2.5.3. BROADCAST RECEIVER	7
2.5.4. ESTRUCTURA DE UN PROGRAMA ANDROID EN DESARROLLO	7
2.5.5. SDK ANDROID	8
2.6. JAVA	10
2.7. JAVASCRIPT OBJECT NOTATION (JSON)	10
2.8. MYSQL	10
2.9. SQLITE	11
2.10. XML	11
2.11. HTML	12
2.12. APACHE	12
2.13. PHP	12
2.14. GEOLOCALIZACIÓN	13

2.15.	FORMAS DE GEOLOCALIZACIÓN EN ANDROID.....	13
2.15.1.	GPS	13
2.15.2.	WIFI	13
2.15.3.	REDES MÓVILES.....	14
2.16.	MENSAJES DE TEXTO (SMS)	14
2.17.	ECLIPSE	14
2.18.	ENCRIPTACIÓN	15
2.19.	OPEN STREET VIEW	15
2.20.	APLICACIONES SIMILARES.....	15
2.20.1.	ADMINISTRADOR DE DISPOSITIVOS ANDROID	16
2.20.2.	PREYPROJECT	18
2.20.3.	CERBERUS.....	21
2.21.	METODOLOGÍA XP (EXTREME PROGRAMMING).....	25
2.21.1.	PLANIFICACIÓN DEL PROYECTO	25
2.21.2.	DISEÑO	25
2.21.3.	CODIFICACIÓN.....	25
2.21.4.	Pruebas.....	26
2.22.	PRUEBAS ALFA	26
2.23.	PRUEBAS DE CAJA NEGRA.....	26
	METODOLOGÍA	27
3.1.	PLANIFICACIÓN DEL PROYECTO	28
3.1.1.	ANÁLISIS DE LOS APLICATIVOS EN EL MERCADO	28
3.1.2.	ANÁLISIS DEL ENTORNO DE DESARROLLO.....	30
3.2.	DISEÑO.....	32
3.2.1.	ANÁLISIS DE LA HERRAMIENTA DE DIAGRAMACIÓN DE CASOS DE USO	32

3.2.2. ANÁLISIS DE LA HERRAMIENTA DE DIAGRAMACIÓN DE PROTOTIPOS	34
3.3. CODIFICACIÓN.....	36
3.4. PRUEBAS.....	37
RESULTADOS Y DISCUSIÓN	38
4.1. PLANIFICACIÓN DEL PROYECTO	38
4.1.1. APLICACIONES ACTUALES EN EL MERCADO	38
4.2. DISEÑO.....	38
4.2.1. CASOS DE USO SMARTPHONE:.....	38
4.2.2. CASOS DE USO SISTEMA WEB	40
4.2.3. DISEÑO DE PANTALLAS PARA LA APLICACIÓN	41
4.2.4. DISEÑO PANTALLAS SISTEMA WEB	44
4.2.5. MODELO DE LA BASE DE DATOS.....	47
4.3. CODIFICACIÓN.....	48
4.3.1. BASE DE DATOS:	48
4.3.2. APLICATIVO ANDROID.....	49
4.3.3. SISTEMA WEB	50
4.4. PRUEBAS.....	52
4.4.1. PRUEBAS DISPOSITIVO	52
4.4.2. PRUEBAS SISTEMA WEB	54
4.4.3. PRUEBAS DE ENVIO PALABRAS CLAVES	55
CONCLUSIONES Y RECOMENDACIONES	63
5.1. CONCLUSIONES	63
5.2. RECOMENDACIONES.....	64
BIBLIOGRAFÍA.....	65

ÍNDICE DE FIGURAS

Figura 1. Imagen del mapa para geolocalización (Google, 2015).....	17
Figura 2. Configuración control web. (PreyProject, 2015).....	20
Figura 3. Resultados localización	20
Figura 4. Pantalla beta prey localización. (PreyProject, 2015).....	21
Figura 5. Localización pantalla (Cerberus, 2015).....	24
Figura 6. Proceso General.....	39
Figura 7. Usuario nuevo registrándose	39
Figura 8. Usuario registrado	40
Figura 9. Recibir acciones al dispositivo	40
Figura 10. Registro usuario nuevo	41
Figura 11. Acciones usuario nuevo registrado.....	41
Figura 12. Pantalla de inicio.....	42
Figura 13. Pantalla configuración del dispositivo.	43
Figura 14. Pantalla registro nuevo usuario	43
Figura 15. Pantalla modificar palabras claves.....	44
Figura 16. Pantalla de ingreso al sistema	45
Figura 17. Pantalla de control	45
Figura 18. Pantalla de registro nuevo usuario	46
Figura 19. Pantalla recuperación clave.....	46
Figura 20. Modelo de la base de datos.....	47
Figura 21. Pantalla de ingreso aplicativo Android.....	52
Figura 22. Pantalla de configuración general.....	53
Figura 23. Pantalla de configuración palabras claves.....	53
Figura 24. Palabras modificadas.....	54
Figura 25. Pantalla de registro	54
Figura 26. Pantalla de Ingreso	55
Figura 27. Pantalla de control.	55
Figura 28. Mensaje de texto	56
Figura 29. Mensaje Web.....	56
Figura 30. Resultado Sonar	57
Figura 31. Mensaje borrar.....	57

Figura 32. Mensaje borrar web	58
Figura 33. Resultado antes de borrar	58
Figura 34. Resultado después de borrar.....	59
Figura 35. Mensaje texto bloquear.....	59
Figura 36. Bloquear mensaje web	60
Figura 37. Pantalla de bloqueo	60
Figura 38. Mensaje localizar texto	61
Figura 39. Resultado y mensaje localizar	61
Figura 40. Elementos para visualizar la fotografía	62
Figura 41. Página Fotografía.	62

ÍNDICE DE TABLAS

Tabla 1. Metodologías de programación.....	27
Tabla 2. Valoración de criterios.....	28
Tabla 3. Análisis de los aplicativos en el mercado.....	29
Tabla 4. Valoración total de criterios.....	29
Tabla 5. Análisis del entorno de desarrollo.	31
Tabla 6. Valoración total de criterios.....	32
Tabla 7. Herramienta de diagramación.....	33
Tabla 8. Valoración total de criterios.....	34
Tabla 9. Herramientas de diagramación de prototipos	35
Tabla 10. Valoración total de criterios.....	36

RESUMEN

El presente proyecto de tesis fue realizado para el desarrollo de un aplicativo que permita el rastreo de dispositivos con sistemas Android, el cual se pueda controlar mediante mensajes de texto y el uso de un panel web. El crecimiento del uso de dispositivos móviles es cada vez mayor, y en estos almacenamos información importante como por ejemplo: contraseñas, documentos, y/o fotografías que en manos equivocadas puede ser un riesgo para nuestra privacidad, por lo cual, tener un control de nuestro dispositivo en caso de pérdida o robo es importante. Se realizó un estudio exhaustivo sobre las herramientas que existen en el mercado, el cual permitió evaluar las ventajas y desventajas de los aplicativos más utilizados. Tomando esta información se levantó los requerimientos para un aplicativo, donde las mejores características de cada uno de los programas analizados sean plasmadas en uno solo. El sistema permitirá tener una interfaz gráfica, tanto del aplicativo para Android como del sistema de control web, para que sean intuitivos y de fácil uso y además de poder controlar mediante un sistema web se pueda manejar mediante mensajes de texto. Con el desarrollo del aplicativo se puede realizar acciones de sonido, bloqueo, borrado de información, tomar fotografías y localizar, lo cual permite encontrar el dispositivo y en caso de pérdida eliminar la información que contenga el mismo.

ABSTRACT

This thesis project was carried out for the development of an application that allows the tracking of devices with Android systems, which can be controlled through text messages and the use of a web panel. The growth of the use of mobile devices is increasing, and in these devices we store important information for example: passwords, documents, and/or photos, that in the wrong hands could be a risk to our privacy that is why is important to take control of our device in case of loss or theft. A through study was made on the tools that exist in the market, which allowed us to assess the advantages and disadvantages of the most commonly used applications. Given this information, it was possible lifted the gathering of requirements for the application, where the best features of each of the programs examined are embodied in the application proposed. The system will have a graphical interface, both for the application of Android and for the web control system. This will enable the application to be intuitive and easy to use. In addition of can be operated via text messages. With the development of the application you can perform actions of sound, lock, clear information, take photographs, and locate, which allows you to find the device and in the event of loss, delete information that it contains.

INTRODUCCIÓN

En la actualidad, los dispositivos móviles inteligentes se han convertido en elementos importantes que contienen datos personales como nombres de usuario, claves de cuentas de correo o redes sociales, fotografías, información y documentos importantes, así como cualquier otro dato que pueda identificar a una persona.

La creación de una aplicación que sea de libre distribución y no permita el acceso a los datos personales es una necesidad, puesto que, mediante el bloqueo al Smartphone o Tablet se accede a eliminar los datos o localizar en caso de una pérdida o robo el dispositivo.

Los dispositivos almacenan información personal que puede ser susceptible en manos de personas desconocidas, motivo por el cual, es indispensable instalar una aplicación que pueda encontrar el equipo o en última instancia borrar la información confidencial para que esta no pueda ser utilizada de forma inadecuada.

El crecimiento de la demanda de aplicaciones antirrobo en todas las plataformas móviles y la excesiva semejanza entre las existentes en el sector, son las principales motivaciones de este proyecto. A pesar de existir diversas opciones en el mercado, el reto que se persigue es ofrecer una aplicación diferente, que contenga las funciones necesarias para que el usuario pueda localizar el dispositivo y de ser necesario eliminar la información en caso de robo.

Este proyecto busca ofrecer geolocalización, no solo para los usuarios comunes, sino también para grupos empresariales o para que los padres tengan conocimiento de la ubicación de sus hijos y poder controlar el uso del dispositivo.

La importancia de la privacidad en la geolocalización radica en mantener la confidencialidad y el control de quién nos hace una petición y qué nos está pidiendo. En muchas ocasiones los usuarios no son conscientes de la información que publican a través de la red ni de los riesgos que esto conlleva, lo cual hace aún más necesaria la gestión de la privacidad en este tipo de aplicaciones.

Para respetar la mención del párrafo anterior, serán implementados los permisos de visibilidad y un sistema de invitaciones que permitirá gestionar la privacidad eficientemente. Además, las aplicaciones actuales solo ofrecen la posibilidad de ver la última posición de cada usuario, mientras que con esta aplicación, se buscará dar un nuevo enfoque más amplio, ofreciendo la posibilidad de ver no solo la última, sino las diez últimas posiciones de un usuario, para poder observar donde ha estado recientemente.

Otra de sus características que la diferencia, es que puede funcionar sin necesidad de una conexión a internet para ciertas acciones, como por ejemplo, a través de mensajes de texto. Para esto las palabras claves serían definidas por el usuario al instalar el aplicativo, lo que lo hace personalizado.

El objetivo es desarrollar una aplicación de rastreo y de ejecución de comandos remotos en caso de robo o pérdida para dispositivos móviles Android.

Para llegar al objetivo es necesario tener en cuenta los siguientes objetivos específicos:

- Desarrollar una aplicación de búsqueda que utilice la geolocalización en caso de pérdida o robo.
- Realizar una aplicación que reciba mensajes de texto para realizar funciones que ayuden a encontrar el teléfono o a proteger la información personal.

- Proporcionar una aplicación web para controlar remotamente el dispositivo para ejecutar funciones y poder encontrar el dispositivo o en caso que sea necesario proteger la información personal.

En conclusión, la creación de este nuevo aplicativo permitirá encontrar dispositivos perdidos o robados mediante mensajes de texto y página web o a su vez eliminar la información considerada como confidencial por parte del usuario.

MARCO TEÓRICO

2.1. APLICACIÓN MÓVIL

Son programas, también llamadas app, las cuales se puede descargar y acceder directamente desde un aparato móvil como por ejemplo un Smartphone o Tablet. Las aplicaciones existen desde hace tiempo, ya estaban en teléfonos Nokia o BlackBerry años atrás.

Actualmente existe una variedad de aplicaciones, pero en los primeros teléfonos, estaban orientadas para mejorar la productividad de las personas, como por ejemplo: alarmas, calendarios, calculadoras y clientes de correo.

Cuando los iPhone ingresaron al mercado se produjo un cambio, generaron nuevos tipos de negocio que hicieron que las aplicaciones sean algo rentable, tanto para desarrolladores como para los mercados de aplicaciones como App Store para productos Apple, Google Play para productos Android y Windows Phone Store para productos Microsoft. (Cuello & Vittone, 2013)

2.2. SISTEMAS WEB

También conocidos como aplicaciones web, son aquellas que se crean e instalan no sobre un sistema operativo, que pueden ser Windows o Linux, sino que están alojadas en un servidor de internet o sobre una intranet, también llamada red local. Su aspecto es muy similar a las páginas web, pero en realidad los sistemas web tienen funcionalidades muy potentes para poder dar respuesta a casos particulares.

Los sistemas Web pueden ser utilizados en cualquier navegador Web (Chrome, Firefox, Internet Explorer, etc.) sin importar el sistema operativo, además no es necesario instalarlas en cada computadora ya que los usuarios se conectan a un servidor donde se aloja el sistema.

Las aplicaciones Web trabajan con bases de datos que permiten procesar y mostrar información de forma dinámica para el usuario.

Los sistemas Web, marcan la diferencia con otros, los cuales son beneficiosos tanto para las empresas que lo utilizan, como para los usuarios que utilicen este tipo de sistema. Estos sistemas se pueden utilizar en cualquier lugar, siempre que el sistema esté alojado en un servidor público, se pueda acceder a él en cualquier momento. (Kendall & Kendall, 2005)

2.3. SMARTPHONE

Es un equipo celular que hace funciones más avanzadas que las de teléfonos corrientes. El término se utilizó cuando salieron terminales con correo electrónico ya preinstalado, configurado y listo para utilizar. Para la opción de correo electrónico se descargaban los mensajes a la bandeja de entrada, pero mediante redes muy pesadas y se limitaba porque tenían muy poco espacio de memoria. Un Smartphone tiene un software más avanzado lo cual hace que tenga un hardware mejor para soportar el nuevo sistema operativo. (Orozco, 2014)

Es un tipo de dispositivo móvil, los cuales se caracterizan por ser de tamaño pequeño para poder ser transportados con facilidad, se conecta con redes inalámbricas, tienen un sistema operativo y aplicaciones.

En español se lo denomina teléfono inteligente, sus funciones son de un teléfono móvil y al mismo tiempo tiene características de un ordenador, también disponen de una pantalla táctil, también puede tener un teclado alfanumérico QWERTY pudiendo ser este físico o virtual, memoria para guardar información, conectividad mediante bluetooth y/o Wifi, también tiene un procesador, memorias RAM y también tiene aplicaciones. (Oliva, Matin, & Manjavacas, 2014).

2.4. SERVICIOS GOOGLE

Google comenzó como un buscador, en el actualidad propone una amplia gama de herramientas y servicios adicionales. Para las empresas y desarrolladores, tiene herramientas de computación en la nube que permite ahorrar dinero y ayuda a las empresas a ser más productivas. (Google, 2015)

2.4.1. GOOGLE CLOUD MESSAGING (GCM)

Es un servicio gratuito para los programadores, que permite enviar datos desde un servidor a las aplicaciones que se desarrollaron para Android. Puede ser un mensaje ligero que permita indicar a una aplicación que tiene una acción por realizar o un mensaje que contenga hasta 4KB de datos de carga.

2.5. ANDROID

Es un sistema operativo inicialmente utilizado solo para Smartphone, al igual que los sistemas IOS (Apple), Symbian (Nokia), BlackBerry OS y Windows.

Actualmente, Android se instala en diferentes dispositivos, como tabletas, televisores, relojes, mini ordenadores, incluso se ha realizado instalaciones en microondas y lavadoras.

Está basado en Linux, que es su núcleo de sistema operativo, es libre y se puede modificar, es gratuito se lo puede descargar y multiplataforma.

Es un sistema operativo libre, esto quiere decir, que para desarrollar aplicaciones e instalar en cualquier dispositivo no se necesita pagar, por lo cual es muy popular entre los fabricantes de Smartphones y desarrolladores, ya que los costos son bajos.

Un desarrollador puede adquirir el código fuente, inspeccionarlo, compilarlo e incluso modificarlo, como hacen muchos fabricantes que personalizan la interface de este sistema operativo. (Robledo & Robledo, 2012).

2.5.1. ACTIVITY

Cuyas siglas en inglés significa actividades, son pantallas que se visualizarán en el aplicativo, parte con la que el usuario interactúa. Se compone de dos partes la parte lógica que se programa en java, y la parte gráfica que contiene todos los elementos que aparecen en la pantalla, se crea en archivos XML. (Martinez, 2015)

2.5.2. SERVICE

Son componentes que se ejecutan sin interfaz gráfica y se ejecutan en segundo plano. Estos pueden realizar cualquier acción. (Martinez, 2015)

2.5.3. BROADCAST RECEIVER

Es un componente que detecta y reacciona ante determinados mensajes o eventos que se generan en el sistema operativo o por otras aplicaciones. (Martinez, 2015).

2.5.4. ESTRUCTURA DE UN PROGRAMA ANDROID EN DESARROLLO

2.5.4.1. Carpeta src:

Dentro de esta carpeta se encuentra los paquetes y el código java, el cual se encarga de ejecutar las Actividades (Activity), y las funciones que se desarrollen.

2.5.4.2. Carpeta res:

En esta carpeta se encuentra los recursos necesarios de una aplicación Android:

- `res/drawable/`: se encuentra las imágenes de nuestra aplicación y se divide dependiente de la resolución del dispositivo en: `drawable-ldpi`, `drawable-dpi` y `drawable-hdpi`.
- `res/raw/`: Contiene archivos de propósito general, en un formato que no sea XML, ejemplo archivos de sonido.
- `res/layout/`: en esta carpeta se encuentran los archivos XML, que son las pantallas que se va a visualizar en nuestra aplicación.
- `res/values/`: contiene los datos que utiliza la aplicación, como los colores, los textos predefinidos, los estilos definidos y dimensiones.

2.5.4.3. Carpeta gen:

Contiene el conjunto de archivos en código Java, que son creados automáticamente cuando se compila el proyecto, para administrar los recursos de la aplicación. En el archivo R se regenera automáticamente

todas las referencias a archivos y valores que se crean, modifican o eliminan en la carpeta anteriormente descrita.

2.5.4.4. Archivo AndroidManifest.xml:

Éste archivo es el más importante en programas Android. Está escrito en XML, en él se encuentra definida la configuración del proyecto, se definen las actividades, Intents, los permisos de la aplicación, bibliotecas, servicios y los recibidores. (Báez, y otros, 2013)

2.5.5. SDK ANDROID

Es un Kit de Desarrollo de Software o en inglés Software Development Kit. Con este kit se puede desarrollar aplicaciones y ejecutar un emulador que se detallará más adelante. Las aplicaciones en Android se desarrollan en Java.

Este se encargará de instalar y ejecutar las versiones que existen actualmente para Android. Cuando Google libera una versión de Android publica el SDK de esa versión y se puede descargar la nueva versión para poder emular un dispositivo con una versión de Android. Esto ayuda al desarrollo de una aplicación, puesto que, es posible adaptar la aplicación a la nueva versión.

Este también contiene un sistema que permite crear y gestionar emuladores así como la documentación para cada versión del sistema operativo. (Subirats, 2014).

2.5.5.1. Android Debug Bridge (adb)

Permite establecer una conexión entre el computador y el Smartphone o el computador y el emulador, permite tener permisos de super usuario. Además, ejecuta comandos o listar el contenido. Con esta herramienta se puede transferir una aplicación o archivos a un dispositivo o al emulador. (Benbourahala, Android 4: Principios del desarrollo de aplicaciones Java, 2013).

2.5.5.2. Dalvik

Es una variación de Java, permite desarrollar aplicaciones y proporciona todas las interfaces necesarias para acceder a funciones del dispositivo, como GPS, mensajes, agendas, etcétera, utilizando el lenguaje Java. (Robledo & Robledo, 2012)

Se considera que es una máquina virtual que permite ejecutar programas desarrollados en Java.

Sacrifica la portabilidad de Java para poder desarrollar aplicaciones con un mejor rendimiento y un consumo mínimo de energía, esto es importante debido las limitaciones que tiene la capacidad de las baterías, también optimiza el uso de memoria y ejecuta varias instancias de la máquina virtual simultáneamente, encargando al sistema operativo el aislamiento de procesos, gestionar memoria e hilos. (Komatineni & MacLean, 2012)

2.5.5.3. Android Asset Packaging Tool (aapt)

Herramienta que crea y analiza archivos apk, se podría decir que este archivo es el instalador de la aplicación para poder instalar en Android. Este archivo se crea y genera automáticamente en Eclipse. (Benbourahala, Android 4: Principios del desarrollo de aplicaciones Java, 2013)

2.5.5.4. Dalvik Debug Monitor Service (ddms)

Herramienta que ayuda para la depuración de aplicaciones. Permite capturar pantallas, ver los hilos en ejecución, revisar el Logcat y verificar la lista de los procesos en ejecución del dispositivo o del emulador, es posible, simular el envío y recepción de mensajes escritos así como la localización. (Benbourahala, Android 4: Principios del desarrollo de aplicaciones Java, 2013)

2.5.5.5. Emulador (Android Virtual Device)

“El emulador es una herramienta que permite simular un teléfono o una Tablet Android. Por lo tanto, le permite desarrollar en Android incluso si no dispone de un teléfono o de una Tablet Android.

También permite crear múltiples tipos de emulador para simular diferentes tipos de teléfonos, de tamaños de pantalla y de resoluciones.” (Benbourahala, Android 4: Principios del desarrollo de aplicaciones Java, 2013).

2.6. JAVA

Es el lenguaje de programación orientado a objetos más utilizado en el mundo, “es rápido, seguro y fiable. Desde portátiles hasta centros de datos, desde consolas para juegos hasta súper computadoras, desde teléfonos móviles hasta Internet, Java está en todas partes” (Java, 2015).

Es una herramienta para desarrollar aplicaciones web, de escritorio, electrodomésticos e incluso para dispositivos móviles. Se ejecuta en entornos heterogéneos que se comunican a través de redes de comunicación.

Una de las ventajas es la independencia de la plataforma, puede ejecutarse en cualquier sistema operativo o cualquier hardware esto significa que se ejecuta en cualquier dispositivo. (Romero Laguillo, 1998).

2.7. JAVASCRIPT OBJECT NOTATION (JSON)

Es un formato liviano utilizado en JavaScript que sirve para el intercambio y la serialización datos. Se lo usa más frecuente en sitios web ya que es más liviano que los archivos XML. Es independiente del lenguaje, pues tiene una particularidad que la información es leída de manera directa por JavaScript.

JSON es una alternativa a XML, y se puede utilizar para comunicar entre distintas tecnologías de programación, describe los datos con una sintaxis dedicada que se usa para identificar y gestionar datos. (Bassi, 2013)

2.8. MYSQL

Es el sistema gestor de base de datos relacional, robusto, rápido y fácil de usar, en un entorno de red, en arquitecturas cliente/servidor. Evita que la información sea redundante o que se repita información, se puede almacenar y organizar los datos los cuales pueden ser manipulados para

poder insertar, modificar o eliminar datos de dicha información. MySql se puede instalar en máquinas con sistema operativo Linux o Windows.

Tiene una versión gratuita y en ciertos casos se puede necesitar adquirir una licencia comercial.

Otra característica es que es compatible con la mayoría de lenguajes de programación, lo que facilita el desarrollo de aplicaciones o sistemas web. (Ramos & Pérez, 2000)

2.9. SQLITE

Es un motor de base de datos, el cual se diferencia de los demás por su tamaño reducido para ser utilizados en dispositivos móviles, otra característica especial es que sus bibliotecas se enlazan con el programa y es empleado mediante las invocaciones a la misma.

Android está incorporado con todas las herramientas que se necesitan para crear y gestionar este tipo de base de datos. La clase auxiliar más relevante es la SQLiteOpenHelper. (Gómez, 2015)

2.10. XML

Es un lenguaje de marcados, para poder crear este tipo de lenguaje es necesario definir un DTD (Document Type Definition). Un DTD especifica que etiquetas van hacer validas, los atributos de la etiqueta, que etiquetas están dentro de otra etiqueta y cuál es el orden de dichas etiquetas.

Un archivo XML debe tener las siguientes características:

- Cada etiqueta debe tener un inicio y un final que los identifique.
- Una etiqueta puede ir dentro de otra etiqueta siempre y cuando la que se encuentra dentro finalice antes de la que le contiene.
- Los valores siempre deben ir entre comillas.
- Existen caracteres que no se pueden incluir dentro de las etiquetas estos son; <, > y “, en vez de estos se puede utilizar <, > y ". (Ramos & Pérez, 2000).

2.11. HTML

No es un lenguaje de programación, más bien es un lenguaje de escritura que comprende etiquetas, siendo estas una instrucción contenida entre corchetes angulares y por lo general contiene también una etiqueta de cierre.

HTML se lo genera en archivos de texto que contendrá las instrucciones o etiquetas y que un navegador web las interpreta. (Sanchez, 2012)

2.12. APACHE

Es un servidor web que es muy popular, ya que es libre y gratuito, el cual se lo puede descargar desde la página web. Una de las mayores ventajas de este servidor es que es multiplataforma, es decir trabaja con distintos sistemas operativos y su rendimiento no tiene problemas.

Desde 1996, tiene una buena acogida, debido a que es estable y seguro. Tiene soporte SSL y TLS, da soporte a los distintos lenguajes de programación. (Vasquez, 2008)

2.13. PHP

Es un lenguaje de programación, creado fundamentalmente como herramienta para el desarrollo de aplicaciones web. Se puede diseñar páginas bajo petición capaces de responder de manera dinámica a las demandas del cliente y se puede automatizar una gran cantidad de tareas. En pocas palabras se podría definir que es un lenguaje interpretado de alto nivel integrado en páginas HTML y que se ejecutan en un servidor.

Aunque actualmente existe una gran cantidad de lenguajes y entornos de desarrollo creados para internet PHP se ha consolidado como uno de los lenguajes más utilizados para el desarrollo de páginas dinámicas.

Esta herramienta de desarrollo tiene características notables; entre otras, es ser software libre para la distribución y es multiplataforma. También es el complemento ideal para Apache en el desarrollo de páginas y sistemas web.

Pero lo más destacable es la sencillez de uso que presenta a los programadores, el aprendizaje es fácil y rápido. (Guitierrez & Bravo, 2005).

2.14. GEOLOCALIZACIÓN

2.15. FORMAS DE GEOLOCALIZACIÓN EN ANDROID

En el sistema Android existen tres formas para ubicar un dispositivo móvil.

2.15.1. GPS

El Sistema de Posicionamiento Global utiliza satélites para la navegación, que permite determinar la posición todo del día y toda la noche. Sea cual sea la ubicación en tierra, mar o aire.

Radica en un conjunto de 24 satélites que giran alrededor de la tierra a unos 20.000 kilómetros de altura, envían señales de radio a su superficie. Un receptor GPS es un aparato que recibe las señales de los satélites, el receptor utiliza señales de radio para calcular su posición con un error a los 100 metros no importa el lugar, hora o condición meteorológica. Además, se puede utilizar el GPS, la velocidad y rumbo de traslado, la hora, el día o incluso estimar el tiempo de llegada.

También se cuenta con estaciones terrestres que ayudan a controlar los satélites. (Correia, 2000)

2.15.2. WIFI

Una red wifi utiliza como medio de transmisión el aire. A diferencia de las redes con cables, las redes wifi permiten moverse y no están limitadas a un espacio físico como sucede con las redes con cables.

Cuando un programa necesita geo localizar mediante wifi, sea en un Smartphone, Tablet o cualquier dispositivo móvil Android que tenga conexión mediante esta red, el aplicativo se comunica con Google Geolocation, Skyhook o la empresa que use, en el caso de los navegadores lo hacen automático, le envía al API una lista completa de las MAC Address de todos los dispositivos routers que estén al alcance de la antena WIFI del

dispositivo, el API coteja esas direcciones con latitudes y longitudes, luego triangula para enviar una latitud y longitud muy cercana a la realidad. (Martínez & Moya, 2005)

2.15.3. REDES MÓVILES

Por su diseño, un dispositivo móvil con chip o tarjeta SIM están en contacto con una o varias torres de telefonía. Mediante un cálculo de triangulación es posible saber la posición del móvil con un pequeño margen de error.

El margen de error depende de la cantidad de torres que existan cerca del lugar donde se encuentra el dispositivo, que dentro de las ciudades es más alto, permitiendo un margen de error de hasta 200 metros. (Hevia, 2011)

2.16. MENSAJES DE TEXTO (SMS)

Fueron creados para ofrecer a los usuarios de celulares un nuevo método de comunicación distinto al ya ofrecido que eran las llamadas telefónicas.

Son mensajes cortos que utilizan la red de telefonía móvil, la misma red que se utiliza para realizar llamadas. Es un sistema de almacenamiento y reenvío, para la transmisión de mensajes desde y hasta teléfonos.

El funcionamiento es el siguiente; una vez que un mensaje se envía desde el dispositivo de origen se almacena en una central, que luego es enviado en seguida al dispositivo de destino. Esto significa que si el dispositivo que receptorá el mensaje no está disponible, el mensaje seguirá guardado hasta que el dispositivo se encuentre en línea. (Mint, 2003).

2.17. ECLIPSE

Es una herramienta de programación de código abierto, que funciona en cualquier plataforma, quiere decir que puede funcionar en sistemas operativos Windows o Linux.

Se maneja bajo la licencia Eclipse Public License, por lo cual es software libre. En el cual se puede diseñar programas de primer nivel, componentes,

Web Service o Sitios Web. También se puede desarrollar aplicaciones para Smartphone o Tablet. (Daum, 2005).

2.18. ENCRIPCIÓN

Es la transformación de un texto y cifrarlo a una forma que no puede ser entendida por personas o programas que no tengan el mecanismo para poder decodificarlo. Esto se lo realiza mediante algoritmos que ayudan a llevar este proceso. Se debe tener en cuenta que este proceso debe ser reversible se utiliza algoritmos codificadores para cifrarlo y algoritmos decodificadores para poder descifrarlos y sean entendibles. (Ferrer, 2001).

2.19. OPEN STREET VIEW

Es un proyecto de colaboración, libre y actualizable para todo el mundo, es decir, por personas con un mismo objetivo, que desinteresadamente pueden ver, copiar, modificar y usar información geográfica de cualquier parte del planeta sin tener restricciones. (OpenStreetMap, 2015).

“el proyecto dirigido a crear y ofrecer datos geográficos libres al mundo. Comenzamos debido a que muchos mapas que se creen que son libres, tienen en realidad restricciones legales o técnicas para su uso, lo cual evita que cualquier persona los pueda utilizar de una manera creativa, productiva o inesperada.” (OpenStreetMap, 2015)

2.20. APLICACIONES SIMILARES

Existen varias aplicaciones similares a la que se pretende desarrollar, de las cuales, se escogió las tres con más descargadas desde Play Store, para realizar un comparativo. Estas son:

- Administrador de dispositivos Android.
- PreyProject.
- Cerberus.

A continuación se detalla las aplicaciones antes mencionadas.

2.20.1. ADMINISTRADOR DE DISPOSITIVOS ANDROID

Es una funcionalidad que Google integra actualmente en el sistema Android, con el objeto de proveer al usuario la posibilidad de controlar remotamente el dispositivo móvil.

Este aplicativo nativo controla remotamente desde un computador las siguientes opciones:

- Localizar el dispositivo Android asociado a una cuenta de Google, mediante GPS o Wifi.
- Restablecer el PIN de la pantalla de bloqueo.
- Borrar todos los datos del teléfono.
- Hacer sonar el dispositivo.

Primero para poder acceder a este servicio se debe dar permisos de Administrador del Dispositivo, el cual se encuentra dentro de configuraciones > seguridad > administrador del dispositivo.

Adicional a esto se debe otorgar los permisos correspondientes que se encuentran en Ajustes de Google, este es el icono que se lo representa comúnmente por un engranaje de color plomo, no confundir con ajustes del dispositivo, se debe marcar las opciones “Ubicar este dispositivo de forma remota” y “Permitir bloqueo remoto y restablecer datos de fábrica”, estas opciones dan permisos a la aplicación para funcionar.

Para poder controlar las opciones antes descritas se debe acceder por un explorador de internet a la siguiente dirección web; <https://www.google.com/android/devicemanager>, en la misma que se deberá ingresar la cuenta de Gmail que está asociada al dispositivo.

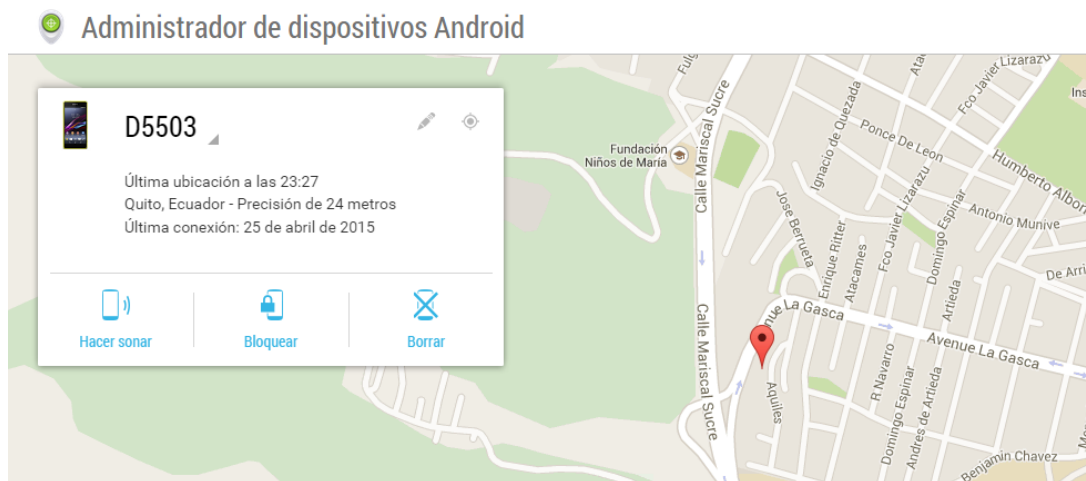


Figura 1. Imagen del mapa para geolocalización (Google, 2015)

Una vez realizado los pasos anteriores saldrá una pantalla en la cual aparece un mapa señalando donde se encuentra el dispositivo como se muestra en la Figura 1, indicando 3 opciones adicionales:

- Hacer sonar: el cual permite que el teléfono emita un ruido para encontrarlo.
- Bloquear: sirve para bloquear el dispositivo con un pin o código.
- Borrar: esta opción elimina todo lo que se tiene en el teléfono, lo dejará como viene de fábrica.
- Geo localizar: visualiza en un mapa el lugar que se encuentra el dispositivo.

Ventajas:

- La página web es muy fácil de utilizar, el usuario puede utilizarlo intuitivamente.
- Cuando se entra a la página web beta automáticamente busca en el mapa la ubicación del aparato electrónico.

Desventajas:

- Estas acciones solo pueden ser realizadas mediante la página web, no se lo puede realizar mediante mensajes de texto.

- Para poder realizar las acciones, se debe contar siempre con una cuenta de Gmail.

2.20.2. PREYPROJECT

Es un aplicativo de monitoreo continuo, lo cual permite mantener un registro de los ordenadores portátiles, Smartphone y Tablet, sin importar el lugar donde se encuentre.

Es software libre, que ayuda a recuperar dispositivos móviles o computadores, funciona en los sistemas operativos más conocidos en el mercado, estos son: Windows, Linux, iOS y Android.

La aplicación es gratuita, pero también dispone de una versión profesional con opciones más avanzadas para usuarios empresariales o que deseen ampliar las opciones de manejo del dispositivo.

El funcionamiento de este aplicativo es registrar el dispositivo, y una vez que se lo instala permanece inactivo, por lo cual no ocupa memoria. Su trabajo empieza una vez que se envíe una señal que activa las funciones, estas pueden ser enviadas mediante una página web o mediante mensajes de texto. Las opciones que se puede realizar con Prey es:

- Geolocalización a través del receptor GPS y la triangulación Wifi.
- Detección del cambio de la tarjeta SIM.
- Bloqueo del móvil o el Tablet.
- Denegación para desinstalar el programa.
- Toma de fotos mediante las cámaras delantera y trasera.
- Ocultar aplicación del menú

Adicionalmente se puede seleccionar en el aplicativo las siguientes opciones:

- Permisos Extras: sirve para que el aplicativo no se pueda desinstalar.
- Desactivar el menú apagar: No permite que el equipo se apague con el botón de encendido/apagado.

Para usar el aplicativo es necesario descargarlo desde la aplicación Play Store de Google, pues este aplicativo no es nativo.

Si no se considera lo primero que se debe realizar es crear una cuenta, una vez realizado el registro, se debe ingresar para que el dispositivo se vincule con la cuenta.

Este aplicativo tiene dos opciones para poder monitorear el dispositivo.

La primera opción es ejecutar las acciones mediante la página web de Prey, <https://panel.preyproject.com/login>. Cuando ya se ingresa a la pantalla principal de Prey, es posible escoger qué dispositivo se desea administrar.

En la página web se encontrarán las siguientes opciones para vigilar el dispositivo:

- Sonar una alarma.
- Enviar un mensaje de información.
- Bloquear el teléfono.
- Hacer que el aplicativo se oculte y no pueda ser visto que está instalado.
- Aparecer un mensaje de error, para que el aplicativo tome una foto.

En la Figura 2 muestra la versión antigua de la pantalla de control, y es la primera que se puede visualizar, la cual permite configurar las opciones de localización, realizar un sonido, bloquear teléfono y el acceso para tomar fotografías.

También tiene una pantalla de reportes, en la cual se puede ver un mapa el cual muestra la ubicación del dispositivo, tal como se muestra en la Figura 3.



Figura 2. Configuración control web. (PreyProject, 2015)

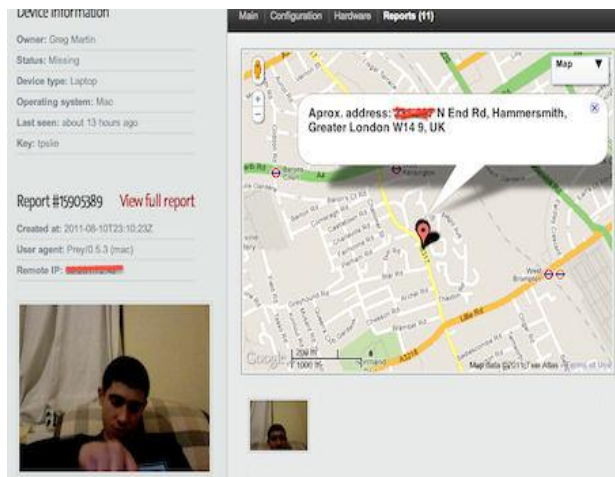


Figura 3. Resultados localización

Actualmente se está trabajando en una pantalla amigable para usuarios con poca experiencia, en la cual se unifica el menú de los teléfonos y la de reportes en una misma pantalla, como se muestra en la Figura 4. (PreyProject, 2015)

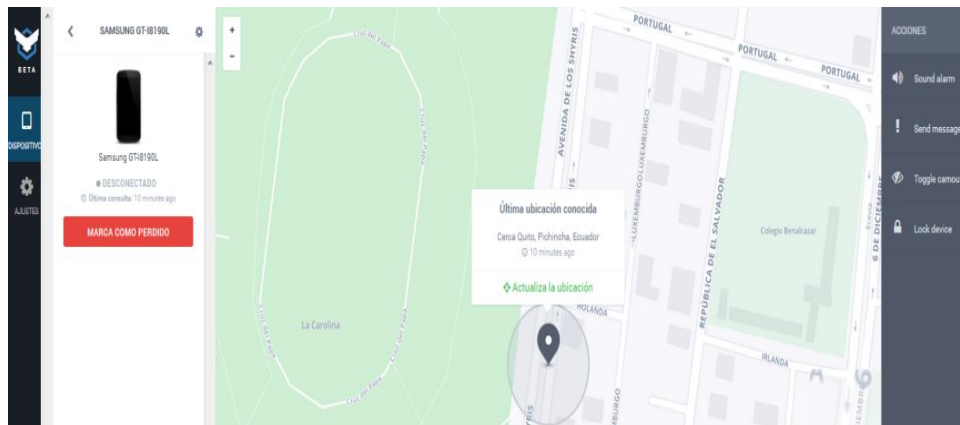


Figura 4. Pantalla beta prey localización. (PreyProject, 2015)

Ventajas:

- La página web que se encuentra en una etapa beta es muy fácil de utilizar, el usuario puede utilizarlo intuitivamente.
- Cuando se entra a la página web beta automáticamente busca en el mapa la ubicación de nuestro teléfono.
- Se puede realizar la búsqueda del equipo mediante mensaje de texto con la palabra “PREY GO”.
- La instalación del aplicativo en el dispositivo es fácil.

Desventajas:

- La página original no es intuitiva y no tiene el mapa
- La página beta todavía tiene problemas cuando se envía una acción.
- En la versión gratuita no se puede realizar cambios a las palabras que activan la geolocalización.
- Solo se puede realizar la ubicación del teléfono cuando se envía la acción por mensajes de texto.

2.20.3. CERBERUS

Este es el aplicativo más completo que existe actualmente en el mercado el cual ayuda a rastrear dispositivos móviles, Smartphone y Tablet.

Lo que se puede realizar con este aplicativo, a través del panel de control web o mediante mensajes de texto desde otro teléfono, es localizar el

dispositivo en el mapa, bloquear el teléfono, mostrar un mensaje en la pantalla, hacer sonar, tomar fotografías, borrar el dispositivo, activar el wifi o plan de datos.

El programa tiene un costo de \$4 dólares americanos, pero también existe la posibilidad de poder probar el aplicativo por un lapso de 7 días. Es de código no es open source.

El aplicativo se lo descarga y se instala en el dispositivo, se registra el usuario y el programa empieza a funcionar en segundo plano hasta que se envíe una acción al celular, y empieza a funcionar, lo que permite un ahorro de batería y de plan de datos.

Estas son algunas de las funciones que se puede realizar mediante la página web y mensajes de texto:

Obtener el historial de ubicación del celular robado.

- Bloquearlo con un código.
- Obtener registro de llamadas y SMS.
- Llamar.
- Enviar SMS.
- Grabar audio.
- Tomar fotografía.
- Capturar vídeo.
- Capturar la pantalla.
- Ocultar aplicación del menú.
- Borrar la memoria interna del teléfono (lo que lo dejaría inutilizable).
- Borrar la tarjeta de memoria.
- Habilitar plan de datos o Wifi.
- Reiniciar el dispositivo siempre que el dispositivo este rooteado.

Hay que registrar los datos, si se tiene una cuenta es necesario dar en el botón "Ya tengo una cuenta, ir al inicio de sesión", caso contrario llenar los datos que pide y le pulsar en el botón "Crear cuenta nueva".

Después hay que elegir “Habilitar administración del dispositivo” y posterior a “Activar” con esto se concede permiso a Cerberus en el dispositivo.

La siguiente opción a elegir es “Configuración SIM checker”, sirve en caso de cambio de chip. Al momento de que ocurra esto se alerta con un mensaje de texto, es posible, poner hasta 3 números telefónicos que serán a los que llegue el mensaje.

Finalmente hay que pulsar sobre “Conceder permisos de Superuser”. En caso de tener root en el teléfono, es factible, darle permisos a Cerberus, con esto se obtiene acceso a algunas opciones extra.

Una vez terminado todo esto se configura la forma en trabajará Cerberus. Las cuales son:

- Cambiar contraseña: cambia la contraseña de la cuenta.
- Cambiar código de marcación: sirve para definir un código numérico para abrir la aplicación y poniendo este código como si realizara una llamada e ingresa en la aplicación Cerberus.
- Change SMS commands keyword: sirve para cambiar la primera palabra que aparece cuando recibe un SMS de alerta de Cerberus.
- Estado: saldrá por defecto “Activado”, en caso de querer desactivar Cerberus, solo hay que poner desactivarlo.
- Limpieza: es para borrar la memoria interna remotamente
- Limpiar tarjeta de memoria: Lo mismo que la opción anterior, pero borrando la tarjeta SD en caso de tenerla.
- External storage: borrar otra ruta o directorio específico.
- Color de pantalla: elegir el color que se pondrá la pantalla cuando se tome una fotografía remotamente.
- Código de desbloqueo incorrecto: Cerberus automáticamente hará una foto si se introduce el código de desbloqueo mal un número X de veces.
- Intentos fallidos: número de intentos fallidos de desbloqueo para hacer la foto.

- Alarma detenida: En caso de estar activado, se hará una foto si el usuario que maneja el teléfono detiene alguna alarma que de Cerberus.
- Mensaje rechazado: La misma opción que la anterior, pero cerrando mensajes de Cerberus.

Para poder administrar y manejar remotamente el dispositivo, hay que acceder a la página de Cerberus, www.cerberusapp.com, e ingresar los datos que se piden. Lo primero que se aprecia es la pantalla de ubicación del dispositivo como se lo puede ver en la Figura 5, que contiene todos los dispositivos que tenemos atados a nuestra cuenta en un combo, también contiene en un combo las acciones que se puede realizar y contiene el mapa de ubicación. (Cerberus, 2015)

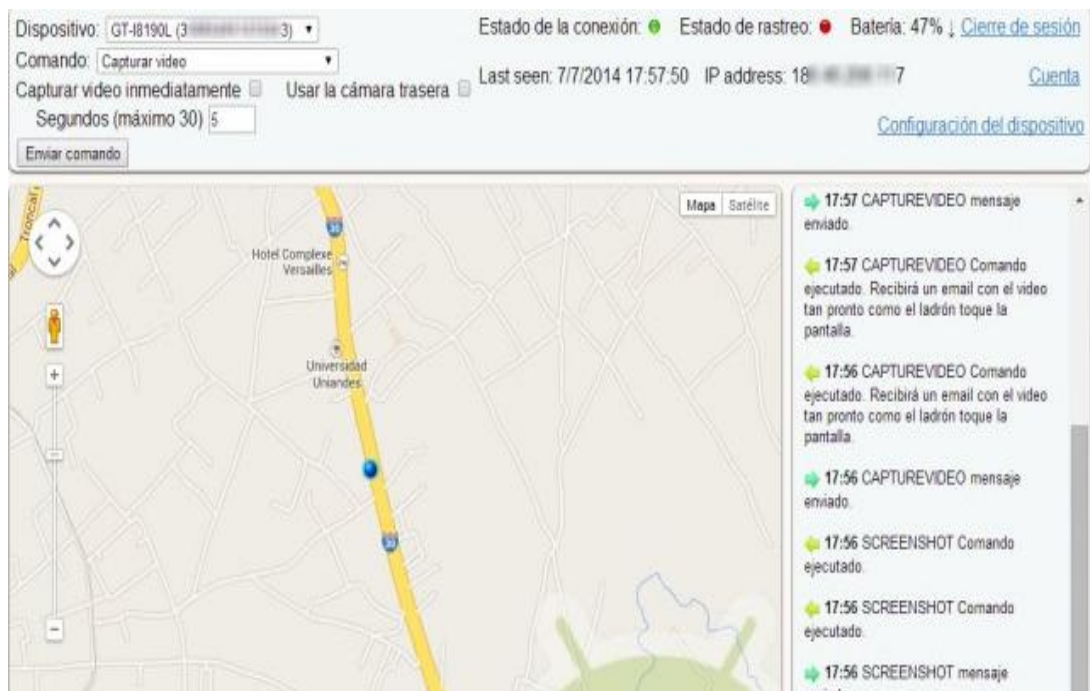


Figura 5. Localización pantalla (Cerberus, 2015)

Ventajas:

- Tiene gran cantidad de acciones para poder realizar al dispositivo.
- Se puede realizar todas las acciones mediante página web o por mensajes de texto.

Desventajas:

- La instalación y la configuración del dispositivo es difícil y tediosa.
- Los mensajes de texto son muy largos.
- Enviar las acciones mediante página web es complicado, puesto que, se tiene que estar escogiendo mediante un combo.

2.21. METODOLOGÍA XP (EXTREME PROGRAMMING)

Es un enfoque para el desarrollo de programas el cual utiliza buenas prácticas de desarrollo. Se basa en principios, prácticas y valores, las cuales son comunicación, simplicidad, retroalimentación y valentía.

Las etapas de esta metodología son:

2.21.1. PLANIFICACIÓN DEL PROYECTO

En esta etapa se comienza a trabajar junto al cliente para decidir y analizar que funcionalidades debería contener el sistema, se lo realiza mediante la información que nos proporciona el cliente.

2.21.2. DISEÑO

Para XP el diseño debe ser lo más sencillo posible, para entender con más facilidad los detalles y poder guiar la implementación de lo conversado con el cliente. De ser necesario se debe rediseñar los diagramas, para llevar un registro de los cambios realizados.

2.21.3. CODIFICACIÓN

Una vez conversado con el usuario y diseñado un trabajo preliminar, el o los desarrolladores realizan cada uno de los diseños en forma de código, una vez terminado cada diseño se realiza las pruebas correspondientes. Una vez terminado y realizado las pruebas se une con los demás módulos que se diseñaron.

2.21.4. Pruebas

Las pruebas unitarias de cada diseño es un elemento clave de la metodología XP. Si es necesario se vuelve a codificar y se realiza las pruebas de los cambios que se realizaron. (Kendall & Kendall, 2005)

2.22. PRUEBAS ALFA

En estas pruebas el cliente realiza las pruebas respectivas del sistema, el o los desarrolladores están presentes pero como observadores, ellos se encargaran de analizar y notar el comportamiento y reacciones del usuario, los errores que puedan existir. (Salvador, 2003)

2.23. PRUEBAS DE CAJA NEGRA

Se lo realiza sobre la interfaz del programa y se pretende concluir que el software funciona correctamente, es decir, que se ingresa información la acepta adecuadamente y se produce una salida correcta. No importa la estructura lógica interna del software. (Alonso, Martínez, & Pérez, 2005)

METODOLOGÍA

Se utilizó la metodología Programación extrema XP para la creación del aplicativo. A continuación se detalla las fases de esta metodología para la creación del aplicativo Android y sistema web.

- **Sistema como algo cambiante:** Es adaptable en caso de cambios en los procesos de desarrollo
- **Colaboración:** Permite la integración de un grupo de trabajo entre el cliente y los desarrolladores.
- **Simplicidad:** Es de fácil uso y de entendimiento.
- **Adaptabilidad:** Se adapta a cualquier desarrollo.
- **Excelencia técnica:** Permite entender con facilidad lo que se requiere.
- **Resultados:** Permite obtener resultados positivos a la entrega del sistemas

Tabla 1. Metodologías de programación

Criterios	Aplicativos		
	RUP	Scrum	XP
Sistema como algo cambiante	3	3	3
Colaboración	1	3	3
Simplicidad	3	3	3
Adaptabilidad	2	2	3
Excelencia técnica	1	1	3
Resultados	2	2	3

Tabla 2. Valoración de criterios

Aplicativo	Valoración total de criterios.
RUP	12
SCRUM	14
XP	18

3.1. PLANIFICACIÓN DEL PROYECTO

Para esta fase se investigó los diferentes aplicativos que existen, se eligió los tres aplicativos más relevantes y se recopiló las mejores funciones que tiene cada uno de los aplicativos y se levantó la información requerida para el desarrollo del sistema.

A continuación se mostrará el análisis para seleccionar los tres aplicativos.

3.1.1. ANÁLISIS DE LOS APLICATIVOS EN EL MERCADO

Para el estudio comparativo de los aplicativos se tomó en cuenta los siguientes criterios:

- **Interfaz de fácil manejo:** Debe ser de fácil uso e intuitivo para el usuario.
- **Descargas:** Numero de descargar que tiene el aplicativo en Google Play.
- **Calificaciones positivas:** Calificaciones que obtuvo positivamente por los usuarios en Google Play.
- **Calificaciones Negativas:** Calificaciones que obtuvo negativamente por los usuarios en Google Play.
- **Control Web:** Debe tener opción para controlar mediante un sistema Web.
- **Actualización:** El tiempo de la última actualización del aplicativo no debe ser muy extenso.

Tabla 3. Análisis de los aplicativos en el mercado

Criterios	Aplicativos				
	Cerberus	encontrar mi teléfono	Find my phone	Prey Anti Robos	Android Device Manager
Interfaz de fácil manejo	2	1	1	3	3
Descargas	3	1	1	2	3
Calificaciones positivas	3	2	1	3	3
Calificaciones Negativas	3	2	1	3	3
Control Web	3	1	1	3	2
Actualización	3	3	1	3	3

Tabla 4. Valoración total de criterios.

Aplicativo	Valoración total de criterios.
Cerberus	17
Encontrar mi teléfono	10
Find My Phone	6
Prey Anti Robos	17
Android Device Manager	17

También se realizó el análisis de cada uno de los componentes que son necesarios para la realización del aplicativo, como por ejemplo los permisos que contiene el sistema Android.

Los permisos principales que se utilizaron para que el aplicativo pueda acceder a funciones del dispositivo fueron:

- **INTERNET:** Permite acceso al internet.
- **READ_SMS:** Permite leer los mensajes.
- **ACCESS_COARSE_LOCATION:** Permite la geolocalización.
- **CAMERA:** Permite acceder a la cámara.
- **READ_EXTERNAL_STORAGE:** Permite leer el contenido de la memoria externa
- **WRITE_EXTERNAL_STORAGE:** Permite escribir o modificar la memoria externa.
- **WAKE_LOCK:** Permite el bloque de la pantalla.
- **CHANGE_CONFIGURATION:** Permite la modificación de las configuraciones del dispositivo.
- **USES_POLICY_FORCE_LOCK:** Permite el bloque forzado de la pantalla.
- **com.google.android.c2dm.permission.RECEIVE:** Permite recibir los mensajes de Google.

Se usó Eclipse como entorno de desarrollo o IDE, pues el SDK Android que se utiliza para este es mucho más avanzado que el de Netbeans. En comparación con Android Studio tiene menos problemas o errores de ejecución.

A continuación se mostrará el análisis de la herramienta de programación o entorno de desarrollo.

3.1.2. ANÁLISIS DEL ENTORNO DE DESARROLLO

Para el estudio comparativo de entornos de desarrollo se tomó en cuenta los siguientes criterios:

- **SDK Android:** Conjunto de librerías actualizadas.
- **Estabilidad:** El entorno de desarrollo no debe contener errores.
- **Curva de aprendizaje:** Aprendizaje de la herramienta en el transcurso del tiempo
- **Facilidad de adopción para desarrolladores:** Facilidad con que la herramienta se pueda adaptar librerías.
- **Comunidad Activa:** Es necesario una comunidad activa para la resolución de problemas.
- **Ejemplos demostrativos:** Es necesario ejemplos los cuales el desarrollar se pueda guiar.

Tabla 5. Análisis del entorno de desarrollo.

Criterios	Entornos de desarrollo		
	Eclipse	Android Studio	Netbeans
SDK Android	3	3	1
Estabilidad	2	2	3
Curva de aprendizaje	3	2	2
Facilidad de adopción para desarrolladores	3	1	2
Comunidad Activa	3	1	3
Ejemplos demostrativos	3	2	2

Tabla 6. Valoración total de criterios.

Entorno de desarrollo	Valoración total de criterios.
Eclipse	17
Android Studio	11
Netbeans	13

3.2. DISEÑO

Se diseñó los casos de uso del aplicativo el cual funcionará en el Smartphone con sistema Android el cual sirvió para recibir los comandos de los mensajes de texto o del sistema web, también se diseñó los casos de uso para el sistema web.

Se plasmó el diseño del aplicativo el cual es intuitivo para el usuario.

Se diseñó el sistema web para controlar remotamente el dispositivo móvil.

Se diseñó la base de datos, en la misma que se llevará la información de los usuarios, dispositivo, parámetros e información de las acciones ejecutadas.

Para la elaboración de los casos de uso se utilizó el programa DIA, que es software libre y multiplataforma, lo que ayudó a crear los flujos sin depender del sistema operativo. También cuenta con una gran variedad de librerías las cuales se pueden descargar de internet.

A continuación se mostrará el análisis de la herramienta de diagramación de casos de uso.

3.2.1. ANÁLISIS DE LA HERRAMIENTA DE DIAGRAMACIÓN DE CASOS DE USO

Para el estudio comparativo de las herramientas de diagramación de casos de uso se tomó en cuenta los siguientes criterios:

- **Multiplataforma:** Se puede instalar independiente del sistema operativo.
- **Permite exportar:** Permite exportar los diagramas a distintos formatos.
- **Curva de aprendizaje:** Aprendizaje de la herramienta en el transcurso del tiempo
- **Facilidad de adopción para desarrolladores:** Facilidad con que la herramienta se pueda adaptar librerías.
- **Comunidad Activa:** Es necesario una comunidad activa para la resolución de problemas.
- **Ejemplos demostrativos:** Es necesario ejemplos los cuales el desarrollar se pueda guiar.
- **Aumentar diagramas:** Se puede obtener más diagramas.

Tabla 7. Herramienta de diagramación.

Criterios	Herramienta de diagramación			
	Dia	Visio	Libre Office Draw	Diagramly
Multiplataforma	3	1	3	3
Permite exportar	3	2	2	2
Curva de aprendizaje	3	3	2	1
Facilidad de adopción para desarrolladores	3	2	2	2
Comunidad Activa	3	3	2	1

Ejemplos demostrativos	3	2	2	1
Aumentar diagramas	3	1	1	1

Tabla 8. Valoración total de criterios.

Entorno de desarrollo	Valoración total de criterios.
Dia	21
Visio	14
Libre Office Draw	14
Diagramly	11

Para la realización de los prototipos de las pantallas del aplicativo y de las pantallas del sistema web se utilizó Ninjamock. Esta herramienta es gratuita y los trabajos que realicen se graban en la web. Es de fácil uso, y aprender a manejar este programa es rápido, es magnífica para ideas básicas. Se puede maquetar para cualquier dispositivo y pagina web.

A continuación se mostrará el análisis de la herramienta de diagramación de prototipos.

3.2.2. ANÁLISIS DE LA HERRAMIENTA DE DIAGRAMACIÓN DE PROTOTIPOS

Para el estudio comparativo de la herramienta de diagramación de prototipos se tomó en cuenta los siguientes criterios:

- **Multiplataforma:** Se puede instalar independiente del sistema operativo.

- **Permite exportar:** Permite exportar los diagramas a distintos formatos.
- **Curva de aprendizaje:** Aprendizaje de la herramienta en el transcurso del tiempo
- **Facilidad de adopción para desarrolladores:** Facilidad con que la herramienta se pueda adaptar librerías.
- **Comunidad Activa:** Es necesario una comunidad activa para la resolución de problemas.
- **Ejemplos demostrativos:** Es necesario ejemplos los cuales el desarrollar se pueda guiar.
- **Soporta distintos prototipos:** Se puede diagramar diferentes aplicativos.

Tabla 9. Herramientas de diagramación de prototipos

Herramienta de diagramación				
Crterios	Ninjamock	Mockingbird	Mockflow	Hotgloo
Multiplataforma	3	3	3	3
Permite exportar	3	2	2	2
Curva de aprendizaje	3	1	1	1
Facilidad de adopción para desarrolladores	3	1	2	1
Comunidad Activa	3	3	3	1

Ejemplos demostrativos	3	2	2	2
Soporta distintos prototipos	3	2	2	3

Tabla 10. Valoración total de criterios.

Entorno de desarrollo	Valoración total de criterios.
Ninjamock	21
MockingBird	14
Mockflow	14
Hotgloo	11

Se utilizó la herramienta, MySQL Workbench, para el modelamiento de la base de datos, la herramienta es compatible con el gestor de base de datos que se utilizó.

3.3. CODIFICACIÓN

En esta fase se desarrolló el sistema, empezando por la implementación de la base de datos la cual debe contener lo necesario para que el usuario pueda ocupar el sistema personalmente.

Luego se continuó con el desarrollo del aplicativo para el sistema Android.

Finalmente se desarrolló el sistema web, el cual sirve para que el usuario controle el dispositivo remotamente.

Con XML se obtuvo la parte visual del aplicativo Android. Es la parte con la que interactúa el usuario.

La utilización del lenguaje JAVA, es necesaria ya que las aplicaciones se programan bajo este lenguaje, y existen muchos foros de ayuda para solucionar problemas o buscar alguna duda que se tenga.

MySQL, se utilizó como gestor de base de datos, porque es multiplataforma y es posible utilizarlo con Android y PHP.

Se ocupó JSON para la transferencia de datos entre Java y PHP, gracias a que es multiplataforma y la mejor forma de comunicar y enviar información entre estos lenguajes.

La utilización de PHP entre aplicativo y servidor, facilitó la comunicación, debido a que son lenguajes compatibles y fáciles de aprender.

PHP funciona desde el servidor, y no desde el navegador.

La herramienta HTML ayudó a realizar la parte visual del sistema de control web.

3.4. PRUEBAS

En esta fase se ejecutó las pruebas alfa y de caja negra para comprobar el correcto funcionamiento del aplicativo y del sistema web.

Cuando se presentó un inconveniente se retornó a la fase de codificación.

Se usó el emulador que viene incorporado con el SDK ANDROID. Éste ayuda a realizar las pruebas sin averiar el dispositivo físico. De esta manera primero se ejecutó las pruebas en el emulador.

Una vez realizadas las pruebas con el emulador y comprobado que el aplicativo no dañará el dispositivo, se realizó las pruebas pertinentes, en el dispositivo físico, que ayudaron a tener resultados más reales.

RESULTADOS Y DISCUSIÓN

4.1. PLANIFICACIÓN DEL PROYECTO

Se utilizó la metodología Programación extrema XP para la creación del aplicativo. A continuación se detalla las fases de esta metodología que muestran los resultados del aplicativo Android y sistema web.

4.1.1. APLICACIONES ACTUALES EN EL MERCADO

A continuación se detallará lo investigado de los actuales aplicativos más utilizados a la fecha de este proyecto, analizando lo mejor de cada una, lo cual permitió realizar la aplicación con las mejores características de cada programa.

Las pantallas del dispositivo se basaron en el aplicativo PreyProject para las pantallas de ingreso, panel de configuración y la configuración de palabras claves y para la pantalla de registros se basó en el aplicativo Cerberus, tomando de cada una de las aplicaciones las formas de distribución y contenido de cada pantalla.

Las pantallas del sistema web se las desarrolló tomando en cuenta el diseño del aplicativo Administrador de Dispositivos Android, por su característica de un mapa completo en la parte central de la pantalla, y del aplicativo Cerberus se tomó como referencia la administración en combos de los dispositivos y acciones.

4.2. DISEÑO

En la Figura 6, se muestra el proceso en general como funciona el sistema. Se puede visualizar las opciones que tiene el cliente tanto en el sistema web y en el aplicativo.

4.2.1. CASOS DE USO SMARTPHONE:

Para el registro del usuario, se debe ingresar los datos necesarios. La Figura 7, muestra cómo será el proceso.

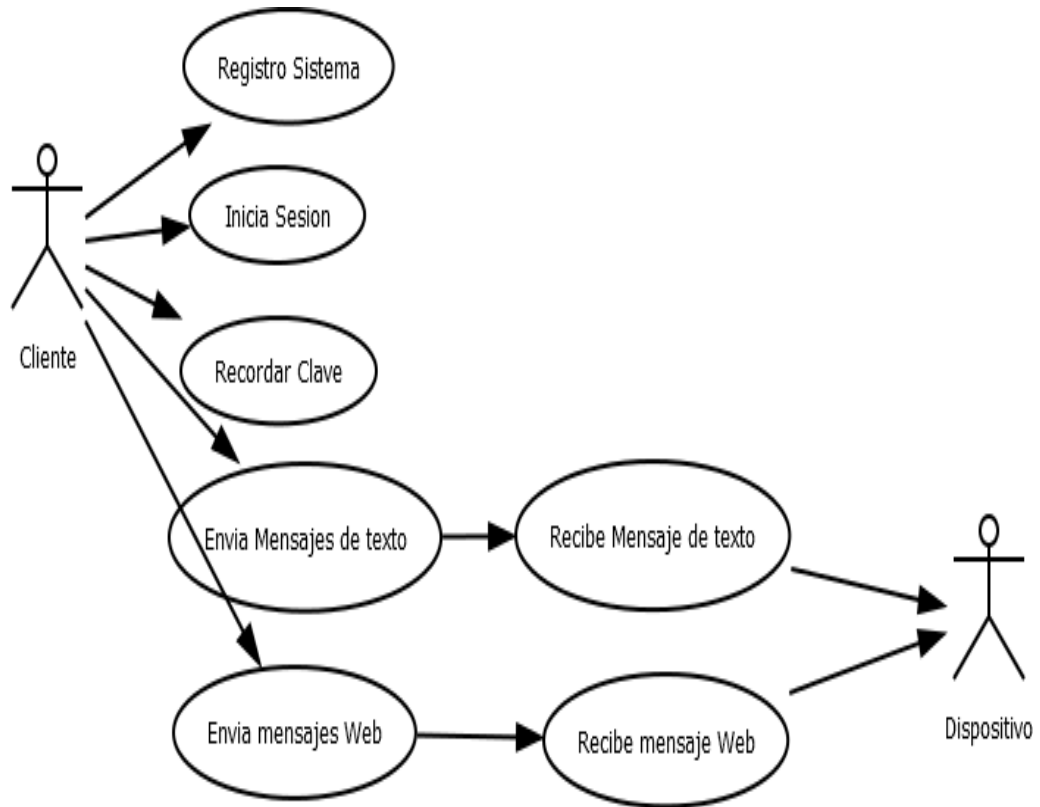


Figura 6. Proceso General.

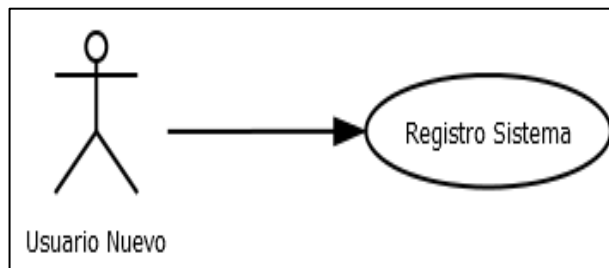


Figura 7. Usuario nuevo registrándose

Un usuario registrado, cuando ingrese con el dispositivo móvil, atará este con su correo electrónico, también podrá seleccionar las palabras claves para activar las acciones. Otra opción es generar una nueva clave en caso de olvidarla. Estos procesos se muestran en la Figura 8.

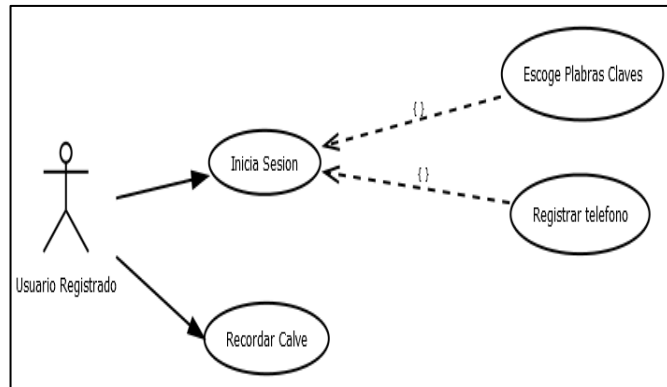


Figura 8. Usuario registrado

La Figura 9, representa la forma en la como el dispositivo obtiene las palabras claves, y estas son ejecutadas según la instrucción que recibió.

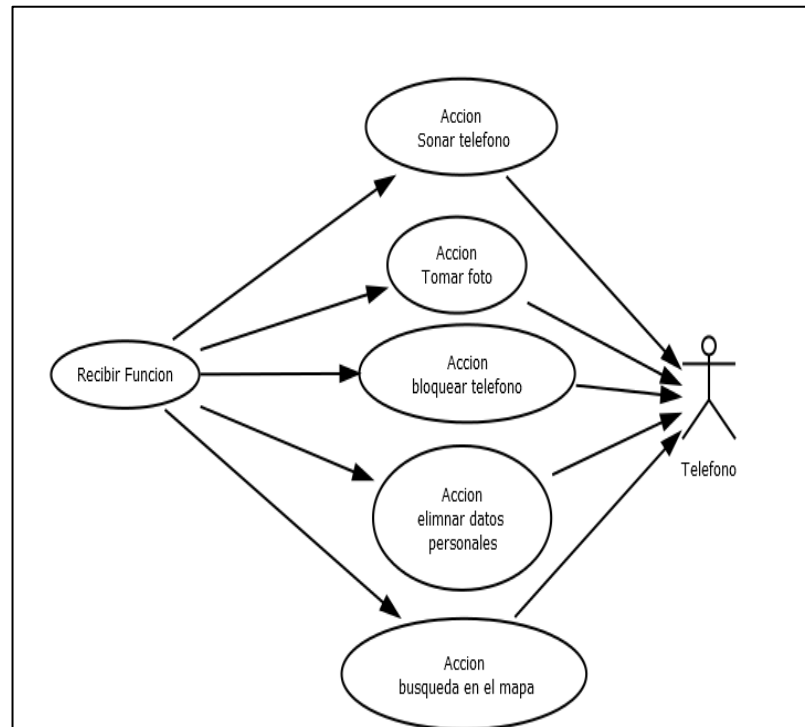


Figura 9. Recibir acciones al dispositivo

4.2.2. CASOS DE USO SISTEMA WEB

Mediante el sistema web el usuario realiza el registro, mas no atará un dispositivo con su correo electrónico, como lo muestra en la Figura 10.

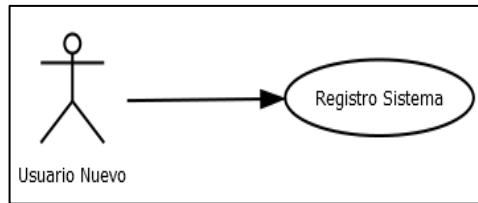


Figura 10. Registro usuario nuevo

Mediante la consola web, se tiene acceso a generar una nueva clave, en el caso de olvidarla.

También se activa las acciones en el dispositivo como lo muestra la Figura 11, estas acciones se enviarán mediante la página web, y google se encargara de enviar los mensajes al dispositivo.

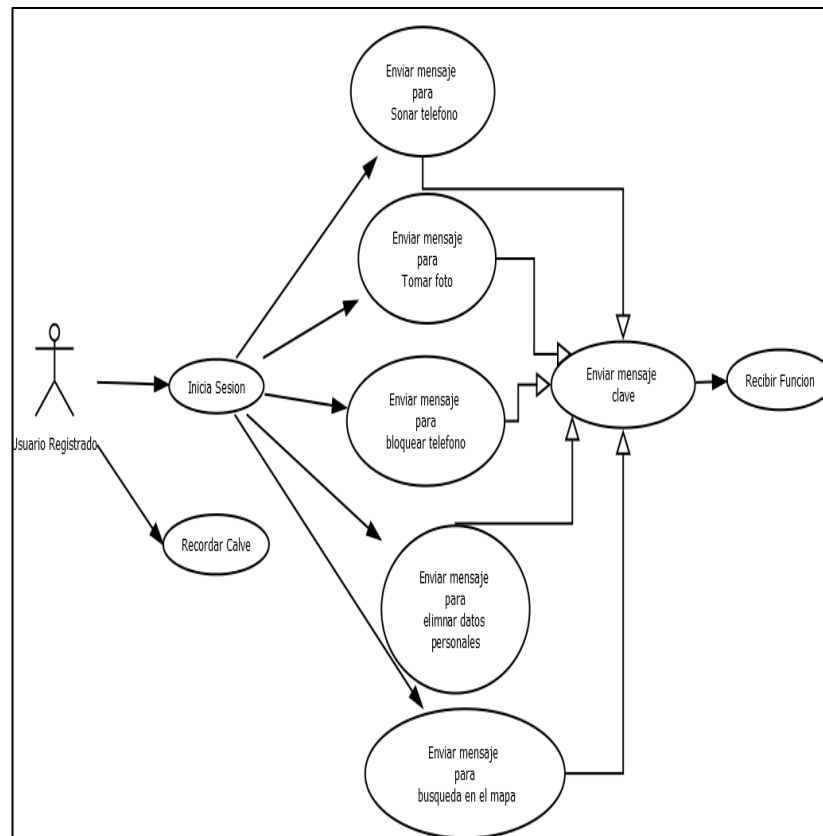


Figura 11. Acciones usuario nuevo registrado.

4.2.3. DISEÑO DE PANTALLAS PARA LA APLICACIÓN

En esta sección se muestra un bosquejo de las pantallas, tanto del aplicativo para Android como las pantallas que tendrá el sistema web.

La pantalla de inicio de la aplicación contiene los campos de texto correo y contraseña, los botones ingreso, registro y olvido de clave, como se muestra en la Figura 12.

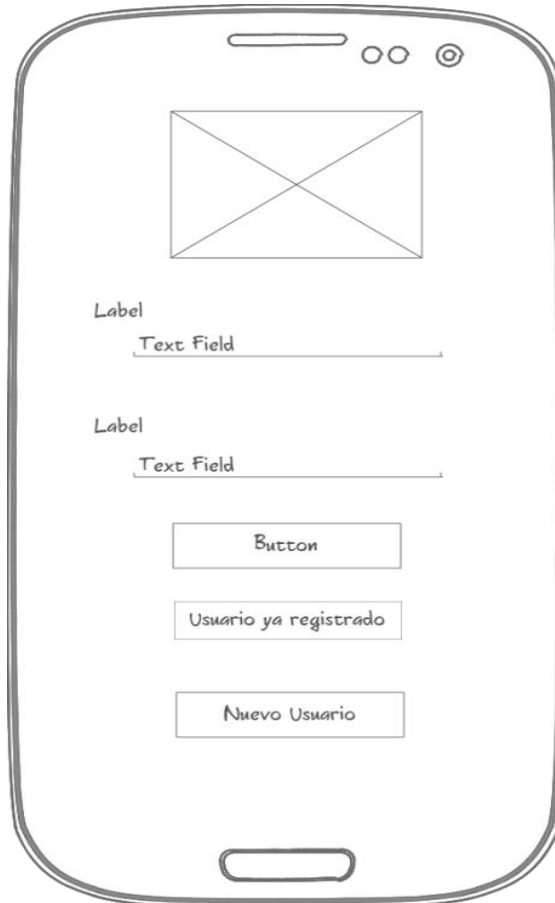


Figura 12. Pantalla de inicio

La Figura 13, muestra cómo se diseñó la pantalla de configuración y los elementos que debe tener como principales, en esta pantalla se tiene las siguientes opciones: administrar dispositivo que envía a la página principal de nuestro sistema web, remover el dispositivo esta opción permite desvincular el dispositivo del usuario, configuración de palabras claves enviará a otra pantalla para dicha acción.

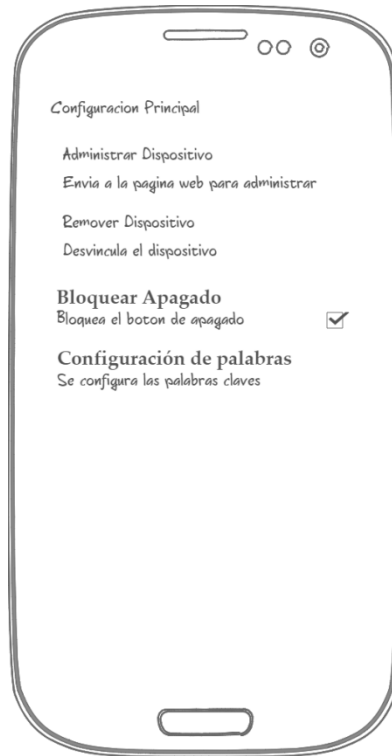


Figura 13. Pantalla configuración del dispositivo.

La pantalla registro tiene: nombre, apellido, correo, y contraseña, tal y como se muestra en la figura 14. También tendrá un botón registro.

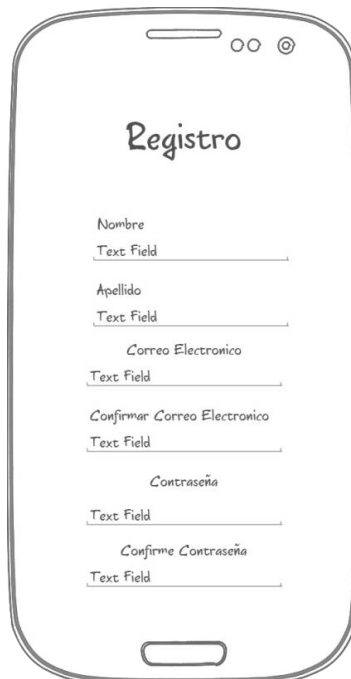


Figura 14. Pantalla registro nuevo usuario

La pantalla de configuración contiene palabras claves las cuales pueden ser modificadas, las palabras de acción no podrán ser cambiadas, tal como se muestra en la figura 15.

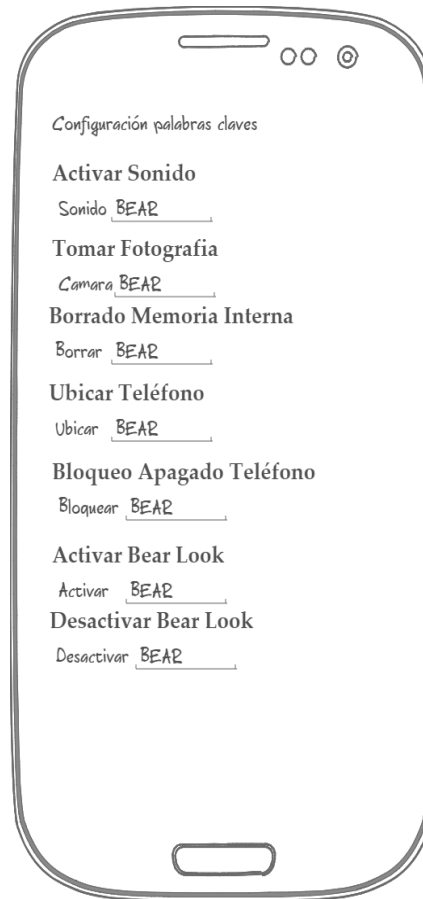


Figura 15. Pantalla modificar palabras claves

4.2.4. DISEÑO PANTALLAS SISTEMA WEB

A continuación se diseñó las pantallas del sistema web.

En la figura 16, muestra cómo se diseñó la pantalla de ingreso al sistema web. Esta pantalla tiene dos cajas de texto que son: de correo y contraseña, además los siguientes tres botones: uno de ingreso, otro que nos enviara a la pantalla de registro de usuario, y otra para generar una nueva clave.



Figura 16. Pantalla de ingreso al sistema

La Figura 17, muestra cómo se diagramó la pantalla que se encarga de enviar las acciones, además contiene el mapa donde se indica la posición del teléfono.

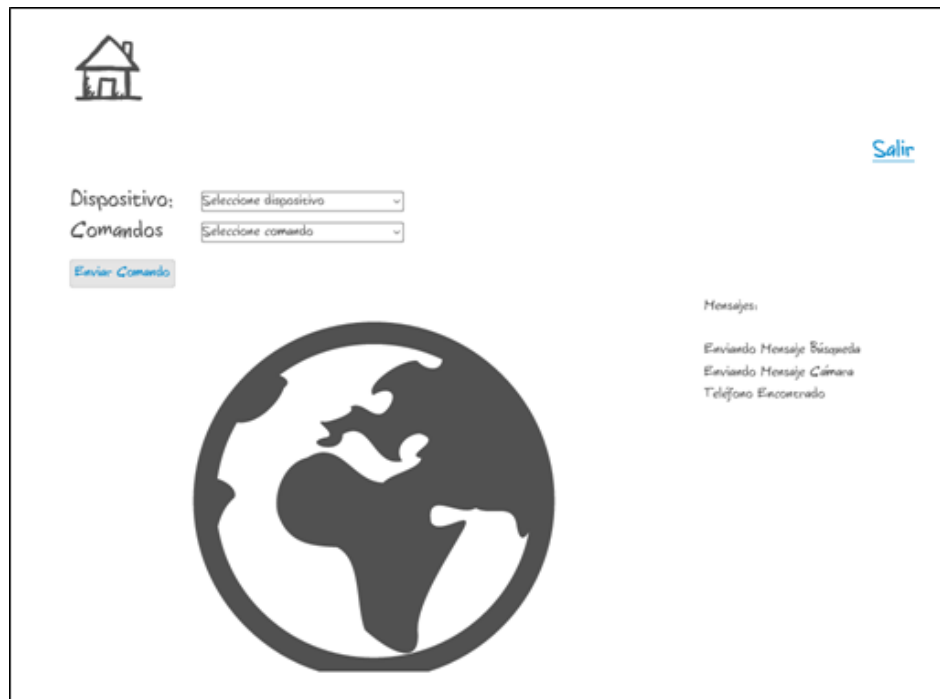


Figura 17. Pantalla de control

En la Figura 18, se puede ver como se maquetó la pantalla de registro de usuario, la cual sirve para crea una cuenta para acceder al sistemas.

La imagen muestra una maqueta de una interfaz web para el registro de un nuevo usuario. En la parte superior izquierda hay un icono de una casa, y en la superior derecha el texto 'BEAR LOOK'. El título principal es 'Ingreso Usuario Nuevo'. Debajo de este título se encuentra un formulario con el encabezado 'Horizontal form'. El formulario contiene los siguientes campos de texto: 'Nombre', 'Confirmar Correo', 'Correo', 'Confirmar Correo', 'Contraseña' y 'Confirmar Contraseña'. A la derecha de cada campo de texto hay un recuadro que indica el tipo de dato que se debe ingresar: 'Nombre', 'Email', 'Email', 'Email', 'Password' y 'Password'. En la parte inferior del formulario hay un botón azul con el texto 'Ingreso'.

Figura 18. Pantalla de registro nuevo usuario

La pantalla de recuperación permite regenerar una clave nueva en caso de olvido, se lo realiza mediante el correo, como se muestra en la Figura 19.

La imagen muestra una maqueta de una interfaz web para la recuperación de una clave olvidada. En la parte superior izquierda hay un icono de una casa. El título principal es 'Olvido de Clave'. Debajo de este título se encuentra un formulario con el encabezado 'Horizontal form'. El formulario contiene un campo de texto etiquetado como 'Correo' con el tipo de dato 'Email' indicado a la derecha. En la parte inferior del formulario hay un botón azul con el texto 'Ingreso'.

Figura 19. Pantalla recuperación clave

4.2.5. MODELO DE LA BASE DE DATOS

La Figura 20 indica el diagrama de la base de datos.

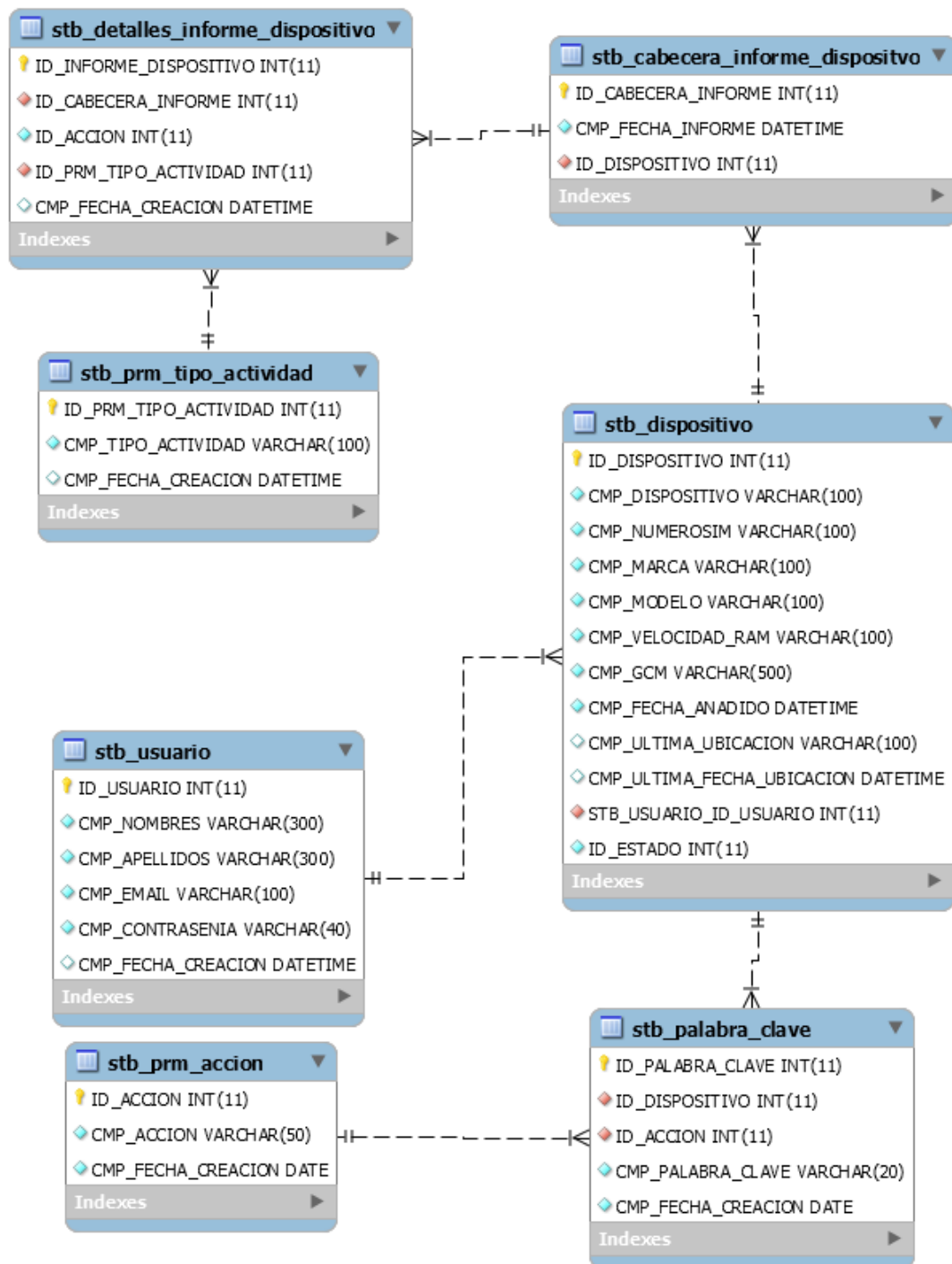


Figura 20. Modelo de la base de datos

4.3. CODIFICACIÓN

4.3.1. BASE DE DATOS:

Para la creación de nuestra base de datos se utilizó el código a continuación descrito.

Para la tabla de usuario, se codifico de la siguiente manera.

```
CREATE TABLE `stb_usuario` (  
  `ID_USUARIO` int(11) NOT NULL,  
  `CMP_NOMBRES` varchar(300) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_APELLIDOS` varchar(300) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_EMAIL` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_CONTRASENIA` varchar(40) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_FECHA_CREACION` datetime(6) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;
```

La tabla donde se almacena los dispositivos es:

```
CREATE TABLE `stb_dispositivo` (  
  `ID_DISPOSITIVO` int(11) NOT NULL,  
  `CMP_DISPOSITIVO` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_NUMEROSIM` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_MARCA` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_MODELO` varchar(100) COLLATE utf8_spanish_ci NOT NULL,  
  `CMP_VELOCIDAD_RAM` varchar(100) COLLATE utf8_spanish_ci NOT  
  NULL,
```

```

`CMP_GCM` varchar(500) COLLATE utf8_spanish_ci NOT NULL,
`CMP_FECHA_ANADIDO` datetime NOT NULL,
`CMP_ULTIMA_UBICACION` varchar(100) COLLATE utf8_spanish_ci
DEFAULT NULL,
`CMP_ULTIMA_FECHA_UBICACION` datetime DEFAULT NULL,
`STB_USUARIO_ID_USUARIO` int(11) NOT NULL,
`ID_ESTADO` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_spanish_ci;

```

4.3.2. APLICATIVO ANDROID

A continuación se muestra el código para que el aplicativo reciba los mensajes de nuestro sistema web.

Archivo RecibirGCM.java

```

public class RecibirGCM extends WakefulBroadcastReceiver{

@Override

    public void onReceive(Context context, Intent intent) {

        if(intent.getStringExtra("message") != null) {

            String mensajeweb = intent.getStringExtra("message");

            cAccion = bdlocalizate.getRegistroAccion(mensajeweb);
            if(cAccion.moveToFirst){

                Accion = cAccion.getString(0);

                mensajeDeshacer = mensajeweb + " " + Accion;
            }
        }
    }
}

```

```

if (mensajeweb.equals("sonar")){
Intent service = new Intent(context, ServicioSonar.class);

    service.putExtra("palabraclave", mensajeDeshacer);

    context.startService(service);

    ServicioLocalizacion.getInstance(context).getLocation();

}

else if (mensajeweb.equals("borrar")){

}

else if (mensajeweb.equals("bloquear")){

}

else if (mensajeweb.equals("foto")){

}

else if (mensajeweb.equals("localizar")){

}

}

```

4.3.3. SISTEMA WEB

Para el ingreso al Sistema web.

```
<?php
```

```
$correo = $_POST['correo'];
```

```
$contrasenia = $_POST['contrasenia'];
```

```
$imei = $_POST['imei'];
```

```
$numerotelefono = $_POST['numerotelefono'];
```



```

require 'funciones_bd.php';

$db = new funciones_BD();

if($db->login($correo,$contrasenia)){
    $resultado[]=array("ingreso"=>"0", "idusuario"=>"0",); }

else{

if($db->existetelefono($imei))

{

if($db->agregardispositivo($db-
usuarioid($correo),$imei,$numerotelefono,$marca,$modelo,$ram,$gcm)) {
$resultado[]=array( "ingreso"=>"1", "idusuario"=>$db->usuarioid($correo),);}

else{$resultado[]=array( "ingreso"=>"0",
"idusuario"=>"0",);}

}

else{

if($db->usuarioid($correo) == $db->idusuariotelefonoexiste($imei)){

$resultado[]=array( "ingreso"=>"-1", "idusuario"=>$db-
>usuarioid($correo),);}

Else

{

$resultado[]=array( "ingreso"=>"-2", "idusuario"=>"-2",);}

}

?>

```

4.4. PRUEBAS

Se realizó las pruebas respectivas de las pantallas y funciones antes mencionadas.

4.4.1. PRUEBAS DISPOSITIVO

Se realizó las pruebas de la pantalla ingreso en el dispositivo. Y se obtuvo como resultado la Figura 21.



Figura 21. Pantalla de ingreso aplicativo Android.

Cuando se ingresó los datos requeridos en la pantalla de ingreso se obtuvo como resultado la pantalla de configuraciones generales, como muestra en la Figura 22.

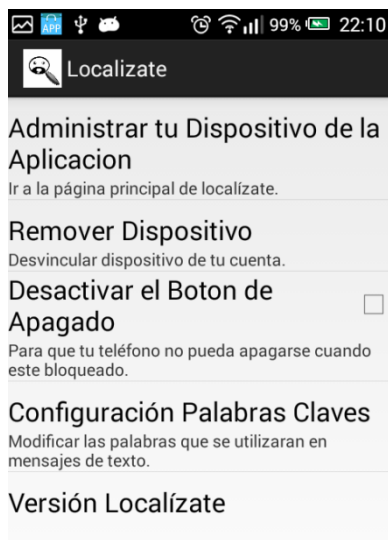


Figura 22. Pantalla de configuración general.

Cuando se pulsa sobre “Configuración Palabras Claves”, se dirige a la pantalla donde se modifica las palabras claves. Por defecto estarán las palabras “localizate” tal como muestra en la Figura 23, las cuales pueden ser modificadas tal como muestra la Figura 24.

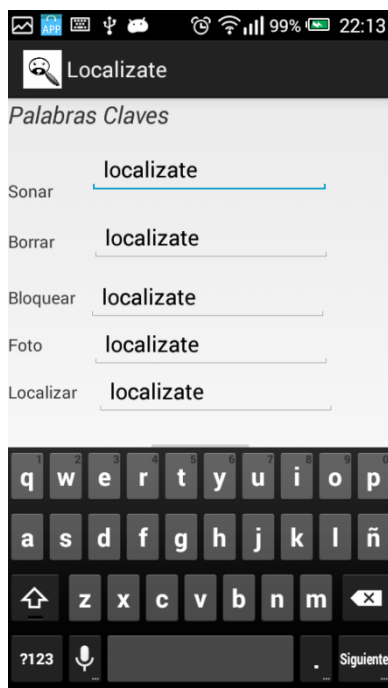


Figura 23. Pantalla de configuración palabras claves.



Figura 24. Palabras modificadas

El aplicativo cuenta también con una pantalla de registro. Esta pantalla se la muestra en la Figura 25.



Figura 25. Pantalla de registro

4.4.2. PRUEBAS SISTEMA WEB

Se realizó pruebas en las pantallas del sistema web obteniendo como resultado lo siguiente.

La pantalla de inicio es la primera pantalla que se podrá observar. El resultado de la pantalla se observa en la Figura 26.

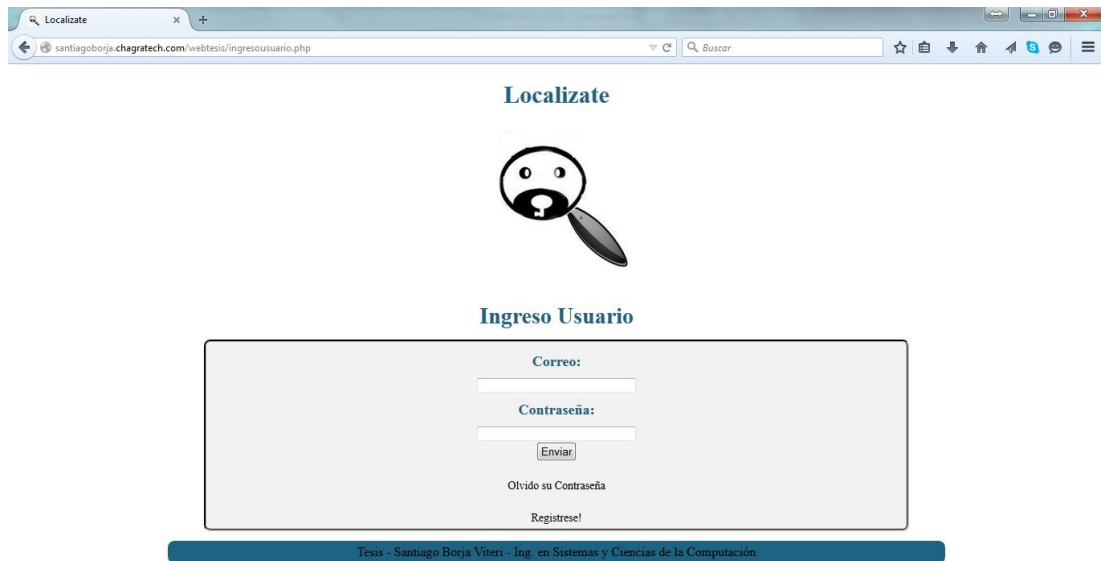


Figura 26. Pantalla de Ingreso

La pantalla de control ingresará, si el usuario digita el correo y contraseña correcta. En la Figura 27 muestra cómo se presenta la pantalla al principio.

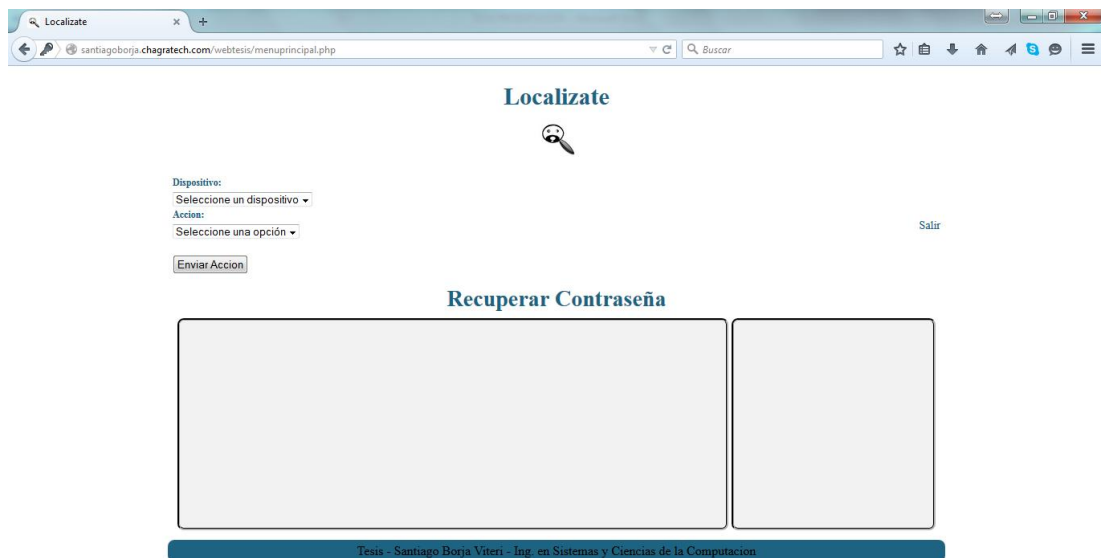


Figura 27. Pantalla de control.

4.4.3. PRUEBAS DE ENVIO PALABRAS CLAVES

Para realizar las acciones se puede ejecutar mediante mensajes de texto o páginas web.

Para la acción sonar se lo realiza mediante mensaje de texto como muestra la Figura 28 y por página web escogiendo el dispositivo y la acción a realizar

como muestra la Figura 29, el resultado que veremos se encuentra en la Figura 30 añadiendo un sonido, el cual se apagará cuando se ingrese la contraseña que es la misma palabra clave.

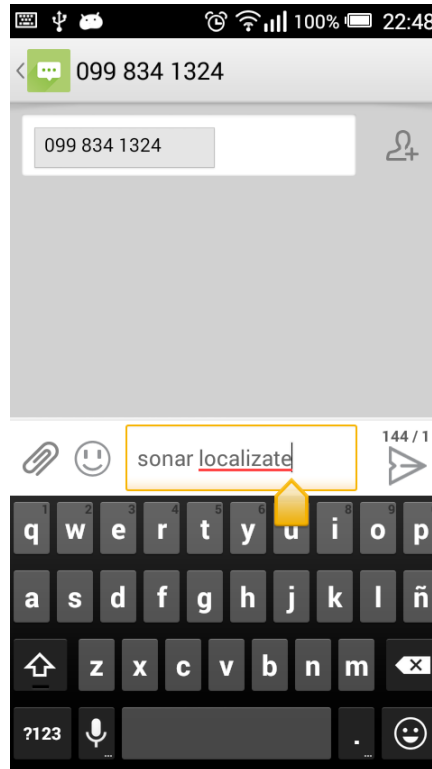


Figura 28. Mensaje de texto

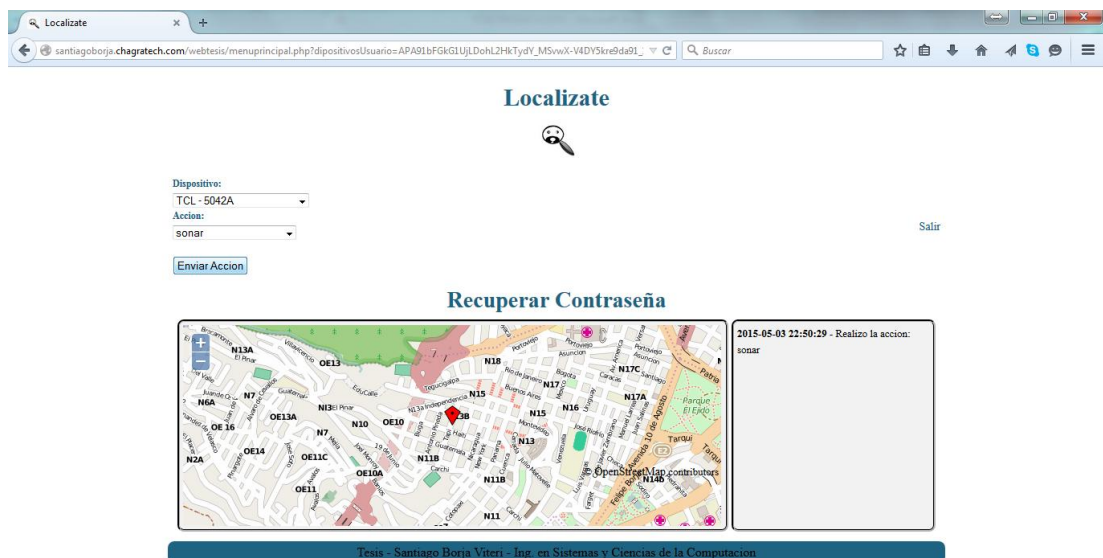


Figura 29. Mensaje Web

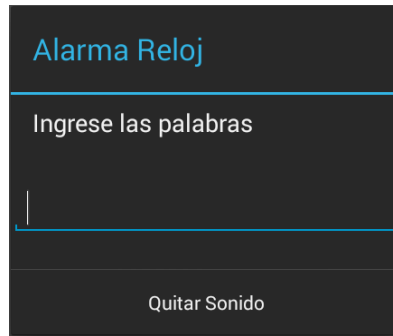


Figura 30. Resultado Sonar

Para la acción borrar se lo ejecuta mediante mensaje de texto como muestra la Figura 31 y por página web escogiendo el dispositivo y la acción a realizar como muestra la Figura 32, el resultado del antes se lo puede visualizar en la Figura 33 y el después en la Figura 34.

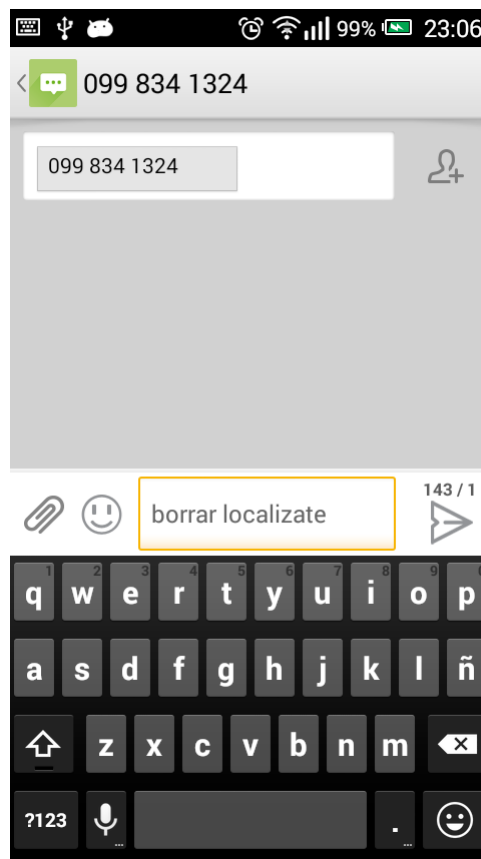


Figura 31. Mensaje borrar

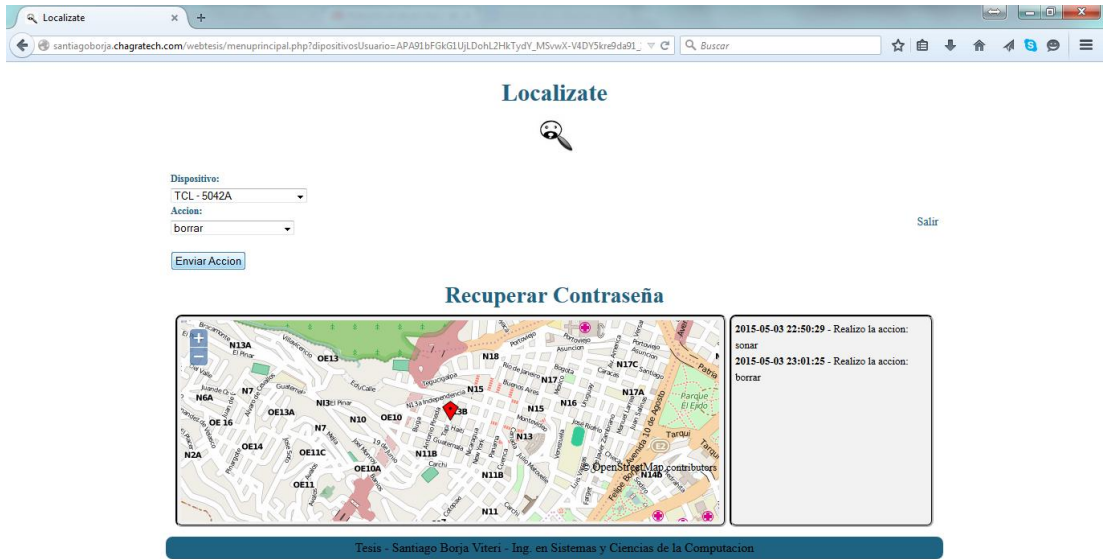


Figura 32. Mensaje borrar web

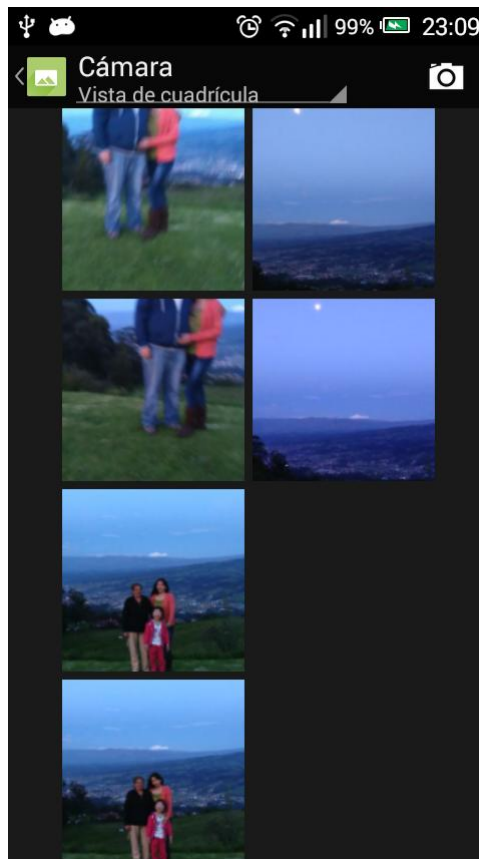


Figura 33. Resultado antes de borrar

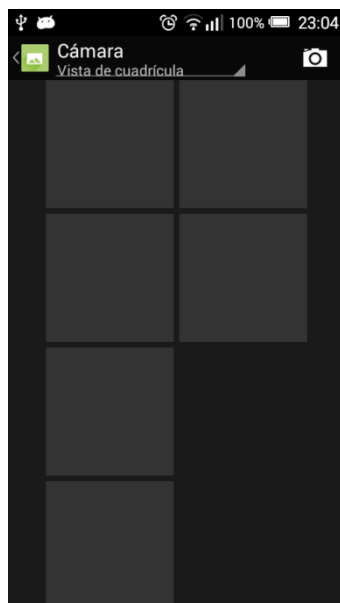


Figura 34. Resultado después de borrar

Para la acción bloquear se lo ejecuta mediante, mensaje de texto como muestra la Figura 35 y por página web escogiendo el dispositivo y la acción a realizar como muestra la Figura 36, la pantalla que bloquea es como se visualiza la Figura 37.

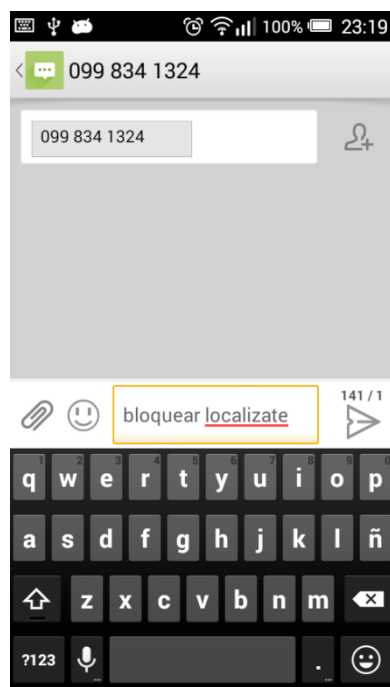


Figura 35. Mensaje texto bloquear

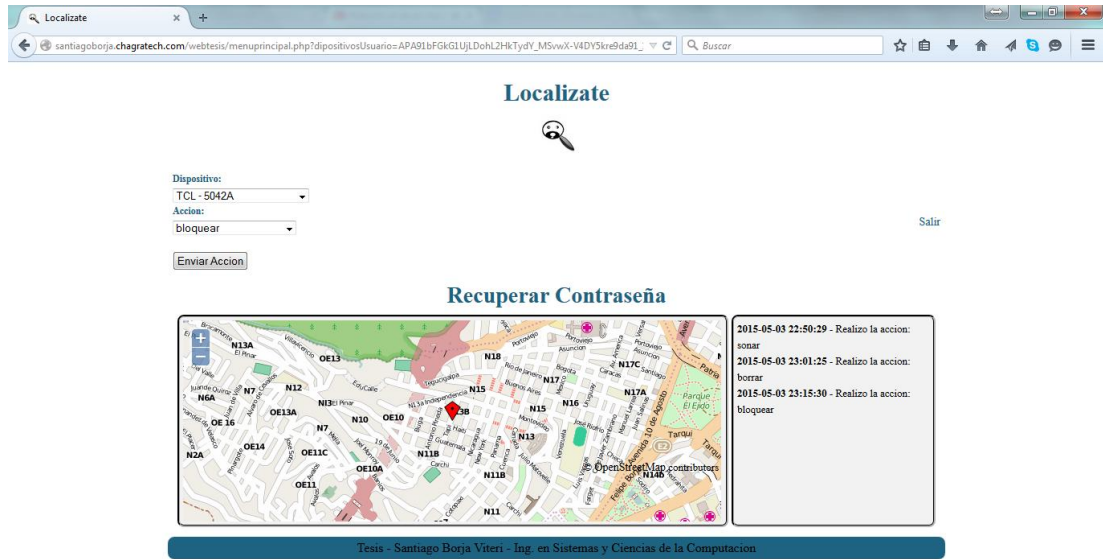


Figura 36. Bloquear mensaje web

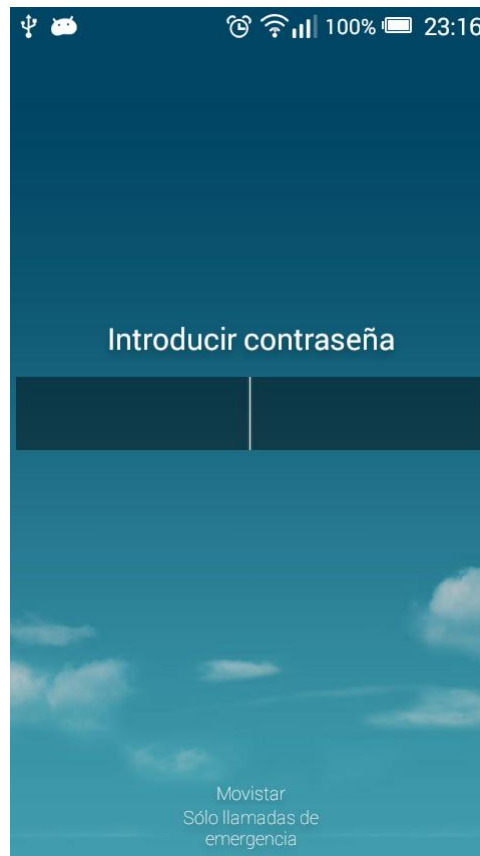


Figura 37. Pantalla de bloqueo

Para la acción de localizar el dispositivo se lo realiza mediante mensajes de texto como indica la Figura 38, mediante página web y el resultado se indica en la Figura 39.

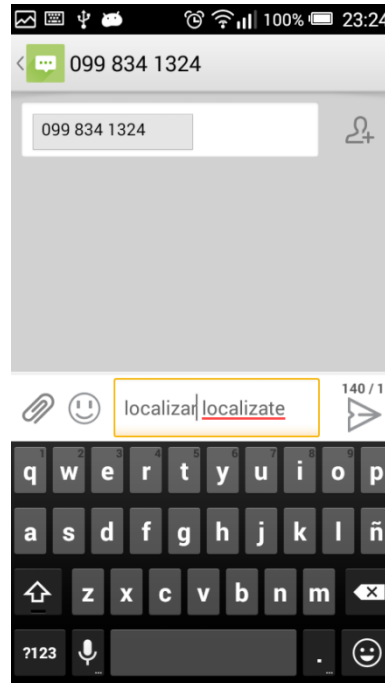


Figura 38. Mensaje localizar texto

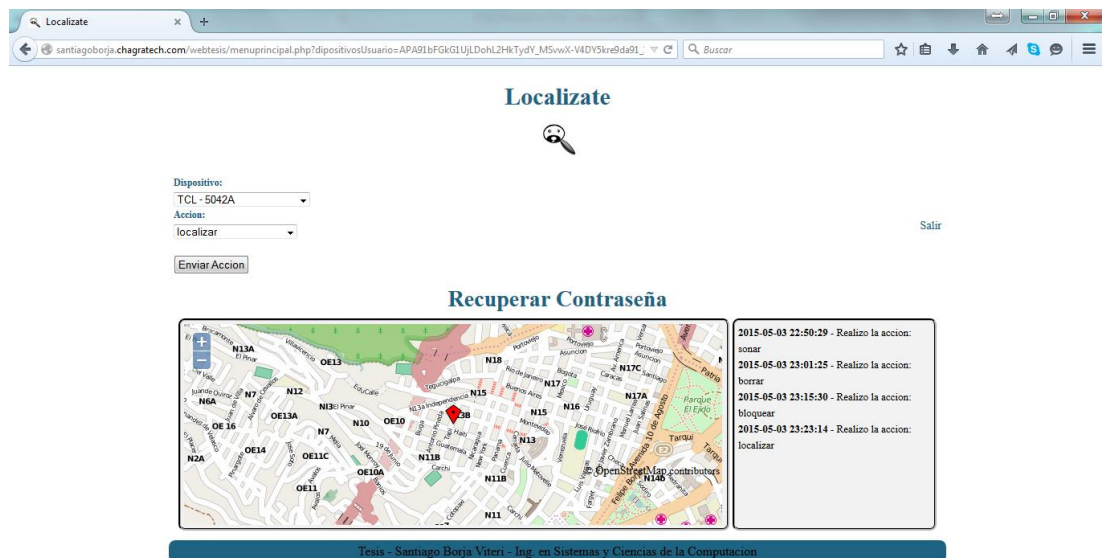


Figura 39. Resultado y mensaje localizar

En la Figura 40, se puede visualizar un combo y un botón “Revisar Foto”, si existe fotografía nos muestra estos elementos, una vez que se escoge la

foto que se desea visualizar nos envía a una pantalla y muestra la fotografía tal como se muestra en la Figura 41.



Figura 40. Elementos para visualizar la fotografía

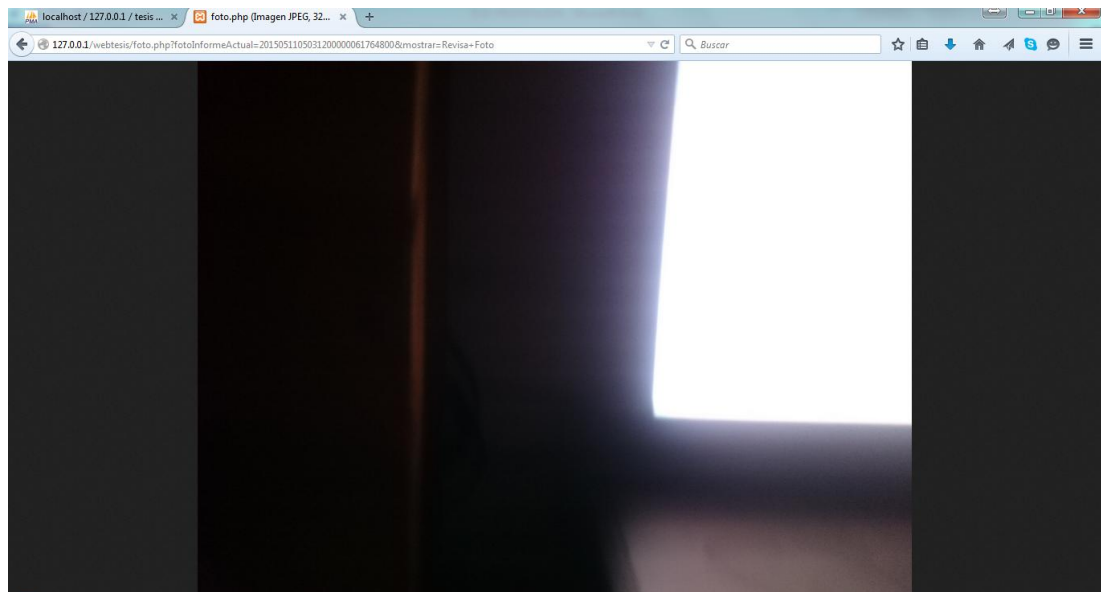


Figura 41. Página Fotografía.

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- La aplicación ayuda a encontrar el dispositivo en caso de pérdida, y en caso de no localizarlo permite eliminar la información de nuestra tarjeta externa.
- El aplicativo recibe los mensajes de texto, y verifica si el mensaje contiene los términos las palabras clave, las cuales se definieron en el momento de configurar el aplicativo.
- El aplicativo realiza las acciones de sonar, bloquear, borrar, localizar y capturar una fotografía, cuando se envía las palabras respectivas y realiza las acciones según el mensaje de texto.
- Mediante este método se envía los mensajes al aplicativo, recibe las palabras claves y realiza las acción sin necesidad de internet, dichas acciones que realiza son: sonar, bloquear y borrar, pues para realizar las funciones de localizar y captura de fotografía se necesita internet. Para recibir los mensajes se necesitó que el dispositivo tenga señal de la operadora telefónica, entre envío y recepción del mensaje tiene un lapso de demora y esto dependió de dicha empresa.
- El sistema web envía los mensajes mediante la plataforma de google GCM. En este sistema web se pude ver la localización de nuestro dispositivo en un mapa y se visualiza un informe de las acciones diarias que se realizaron a nuestro dispositivo.
- Él envío de mensajes por la plataforma web nos permite agilizar las acciones, ya que es más rápido y se depende de los servicios de google los cuales son de respuesta inmediata. Pero para el funcionamiento de este método se necesita estar conectados a un plan de datos o una señal Wifi.
- El aplicativo es intuitivo para el usuario, la configuración es rápida.
- El sistema web permite un rápido acceso para el envío de las acciones.

5.2. RECOMENDACIONES

- El actual sistema se puede mejorar, incluyendo al sistema web con la metodología de responsive design, la cual permitiría una mejor visualización desde cualquier dispositivo.
- La realización de una confirmación de correo, ayudaría a que el usuario tenga la seguridad de que utilizó el correo adecuado.
- Se recomienda la creación de un servicio, el cual permita el envío de mensajes de texto en caso de un cambio de chip.
- Se recomienda evaluar la creación de un servicio el cual permita el envío de las coordenadas mediante mensaje de texto, en caso de no existir una conexión a internet.
- Se recomienda la creación de una pantalla de modificación de datos personales del usuario.

BIBLIOGRAFÍA

- Alonso, F., Martínez, L., & Pérez, F. (2005). *Introducción a la ingeniería del software*. Zaragoza: Grefol S. A.
- Báez, M., Borrego, Á., Cordero, J., Cruz, L., González, M., Hernández, F., . . . Zapata, Á. (2013). Introducción a Android. En M. Báez, Á. Borrego, J. Cordero, L. Cruz, M. González, F. Hernández, . . . Á. Zapata, *Introducción a Android* (págs. 9-10). Madrid: E.M.E. Editorial.
- Bassi, S. (2013). *Python en 8 clases: Aprendiendo a programar con Python*. Buenos Aires: Genes Digitales.
- Benbourahala, N. (2013). *Android 4: Principios del desarrollo de aplicaciones Java*. Barcelona: Editions ENI.
- Cerberus. (22 de 05 de 2015). *Cerberus*. Obtenido de Cerberus: <https://www.cerberusapp.com/ayuda.php>
- Correia, P. (2000). *Guía práctica del GPS*. Paris: Editions Eyrolles.
- Cuello, J., & Vittone, J. (2013). *Diseñando apps para móviles*. Barcelona: José Vittone.
- Daum, B. (2005). *Eclipse 3 para desarrolladores Java*. Madrid: Anaya Multimedia.
- Ferrer, O. (2001). *Telemedicina*. Madrid: Panamericana.
- Gómez, S. (16 de 01 de 2015). *sgolive*. Obtenido de sgolive: <http://www.sgoliver.net/>
- Google. (22 de 04 de 2015). *Google*. Obtenido de Google: <https://support.google.com/accounts/answer/6160491?hl=es>
- Google. (22 de 04 de 2015). *Google*. Obtenido de Google: <http://www.google.com.ec/intl/es-419/about/company/products/>

- Guitierrez, A., & Bravo, G. (2005). *PHP5 a través de ejemplos*. Mexico: RA-MA Editorial.
- Hevia, A. (14 de 11 de 2011). *Xataka on*. Obtenido de Xataka on: <http://www.xatakaon.com/tecnologia-de-redes/como-localizar-mi-posicion-gps-a-gps-wifi-y-redes-gsm>
- Java. (24 de 04 de 2015). *Java*. Obtenido de Java: https://www.java.com/es/download/faq/whatis_java.xml
- Kendall, K., & Kendall, J. (2005). *Análisis y diseño de sistemas*. Mexico: Pearson Educacion.
- Komatineni, S., & MacLean, D. (2012). *Pro Android 4*. APRESS.
- Martínez, D., & Moya, J. (2005). *Comunicaciones en redes WLAN: WiFi, VoIP, multimedia, seguridad*. Madrid: Creaciones Copyright.
- Martinez, E. (22 de 04 de 2015). *udemorelia*. Obtenido de udemorelia: http://www.academia.edu/3439614/Programaci%C3%B3n_de_Dispositivos_Moviles_III_Android_
- Mint, D. (2003). *Servicio de Mensajes Cortos (SMS) el Mercado Telefónico de España*. Oregon: The Wilcox Trading Company.
- Moreno, S. (30 de 03 de 2014). *kustruki*. Recuperado el 27 de 11 de 2014, de kustruki: <http://www.kustruki.com/cerberus-tutorial-asegurar-telefono/>
- Oliva, J., Matin, P., & Manjavacas, C. (2014). *Montaje y mantenimiento de equipos*. Madrid: Ediciones Parainfo.
- OpenStreetMap. (22 de 04 de 2015). *OpenStreetMap*. Obtenido de OpenStreetMap: <https://www.openstreetmap.org/help>
- Orozco, D. (06 de 10 de 2014). *concepto definicion*. Obtenido de concepto definicion: <http://conceptodefinicion.de/smartphone/>

- PreyProject. (22 de 04 de 2015). *PreyProject*. Obtenido de PreyProject: <https://help.preyproject.com/articles/47-what-do-the-different-statuses-mean-ready-unreachable-away-etc>
- Ramos, I., & Pérez, M. D. (2000). *Ingeniería del software y bases de datos: tendencias actuales*. Castilla: Universidad de Catilla-La Mancha.
- Robledo, D., & Robledo, C. (2012). *Programación en Android*. Madrid: Aula Mentor.
- Romero Laguillo, L. F. (1998). *Publicar en Internet: guía práctica para la creación de documentos HTML*. Cantabria: Universidad Cantabri.
- Salvador, J. (2003). *Ingeniería de proyectos informáticos: actividades y procedimientos*. Madrid: Graphic Group.
- Sanchez, M. (2012). *Manual de Desarrollo Web basado en ejercicios y supuestos prácticos*. Malaga: CreateSpace.
- Subirats, J. (03 de 04 de 2014). *fandroides*. Obtenido de fandroides: <http://www.fandroides.com/que-es-y-para-que-sirve-el-sdk/>
- Vargas, M. (02 de 03 de 2013). *enchufadroid*. Recuperado el 27 de 11 de 2014, de [enchufadroid: https://enchufadroid.wordpress.com/2012/03/02/tutorial-cerberus/](https://enchufadroid.wordpress.com/2012/03/02/tutorial-cerberus/)
- Vasquez, J. (2008). *Consulta y actualización de base de datos mediante equipos móviles*. Medellin: ITM.
- Vazquez, A. (2011). *Android: Programación de dispositivos móviles a través de ejemplos*. Barcelona: marcombo.