



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA
ALERTAR UNA EMERGENCIA DE PERSONAS VULNERABLES SOBRE
UN TELÉFONO INTELIGENTE

AUTOR

Erick Alexander Cachipundo López

AÑO

2019



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN MÓVIL PARA
ALERTAR UNA EMERGENCIA DE PERSONAS VULNERABLES SOBRE UN
TELÉFONO INTELIGENTE

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de Ingeniero en Redes y
Telecomunicaciones

Profesor guía

Mg. Iván Ricardo Sánchez Salazar

Autor

Erick Alexander Cachipundo López

Año

2019

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo, Diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente, a través de reuniones periódicas con el estudiante Erick Alexander Cachipundo López, en el semestre 201920, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.”

Iván Ricardo Sánchez Salazar

Magíster en Calidad, Seguridad y Ambiente

C.C. 1803456142

DECLARACIÓN DEL PROFESOR CORRECTOR

"Declaro haber revisado este trabajo, Diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente, del estudiante Erick Alexander Cachipundo López, en el semestre 201920, dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación"

Milton Neptalí Román Cañizares

Magíster en Gerencia de Redes y Telecomunicaciones

C.C. 0502163447

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Erick Alexander Cachipundo López

C.C. 1718546920

AGRADECIMIENTOS

Agradezco a Dios por darme la fortaleza y sabiduría necesaria para culminar una etapa más de mi vida y en especial a mis padres, quienes, con su trabajo y apoyo incondicional hicieron todo lo posible para alcanzar este logro, así como también doy las gracias a mis profesores, amigos y compañeros por sus conocimientos impartidos y en especial a los señores Mg. Iván Sánchez y Mg. Milton Román por guiarme en este proyecto de titulación.

DEDICATORIA

Este proyecto le dedico a mis queridos padres, quienes están conmigo en los momentos buenos y difíciles del camino, me han enseñado que en la vida nada es imposible ya que con trabajo, sacrificio y perseverancia se logra todo lo que uno se propone.

RESUMEN

En lo que corresponde a la presente investigación en el proyecto de diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente, mediante estos dispositivos se puede notificar incidentes de robos a personas en general (tercera edad, capacidades especiales, etc.), así como también de accidentes de las mismas al encargado de seguridad y a una plataforma alojada en un servidor web mediante el uso de servicios web desde el dispositivo. La alerta de emergencia es activada por parte del usuario mediante dos métodos: el primero, al aplastar el botón de pánico en la pantalla principal de la aplicación móvil y el segundo, mediante botones físicos del teléfono inteligente, en este caso la tecla de subir el volumen.

Una vez que el usuario realice cualquiera de las dos acciones, se obtendrá como resultado lo siguiente:

1. Se enviará una alerta con la información básica del usuario al servidor web con los siguientes datos: usuario (nombre, cédula, tipo de sangre y discapacidad), contacto de emergencia, fecha y hora de la alerta y finalmente su ubicación GPS.
2. Se enviará un mail al contacto de emergencia o encargado de seguridad, cuyo contenido son los datos de la persona que está en peligro, así como su ubicación.
3. Y finalmente, se grabará automáticamente un video, mismo que se almacenará en un servidor de Firebase Storage.

Palabras clave: Android Studio, Java, JSSE, Nube (BDD, Firebase y Servidor Web).

ABSTRACT

In what corresponds to the present investigation, in the project of design and implementation of a mobile application to alert an emergency of vulnerable people in a smartphone, it is important to mention that through these devices incidents of robberies can be reported to all people in general (elderly people, special abilities, etc.), as well as their accidents for the security officer and for a platform hosted on a web server through the use of web services from the device. The emergency alert is activated by the user by two methods: the first, crushing the panic button on the main screen of the mobile application and the second, using the physical buttons of the smartphone, in this case, the key to upload the volume.

Once the user performs any of the two actions, the following will be obtained:

1. An alert with the basic information of the user will be sent to the web server with the following data: user (name, identification, blood type and disability), emergency contact, date and time of the alert and, finally, its location GPS.
2. An email will be sent to the emergency contact or security officer, whose content is the data of the person in danger as well as their location.
3. And finally, a video will be recorded automatically, which will be stored on a Firebase storage server.

Keywords: Android Studio, Java, JSSE, Nube (BDD, Firebase y Web Server).

ÍNDICE

1. Capítulo I. Introducción.....	1
1.1. Antecedentes.....	1
1.2. Alcance	3
1.3. Justificación	3
1.4. Objetivo General.....	5
1.5. Objetivos Específicos	5
1.6. Metodología	5
1.6.1. Fases del modelo.....	6
1.6.2. Los métodos de investigación.....	7
2. Capítulo II. Marco Teórico.....	8
2.1. Estadísticas de discapacidades	9
2.1.1. Personas con discapacidad registradas.....	9
2.2. Situación sociodemográfica del adulto mayor	9
2.2.1. Población Adulto Mayor	9
2.3. Fundamentación teórica	12
2.3.1. Sistemas de Información.....	12
2.3.2. Base de datos	12
2.3.3. Arquitectura Cliente – Servidor	13
2.3.4. Lenguaje de programación	14
2.3.5. Aplicaciones móviles.....	14
2.3.6. Sistemas Operativos móviles.....	14
2.3.7. Sistemas Operativo Android	15
2.3.8. Características de Android.....	15
2.3.9. Arquitectura de Android	18
2.3.10. Android Studio	22
2.3.11. Programación sobre Android	24
2.3.12. Componentes de las aplicaciones	25
2.3.13. Streaming	30

2.3.14. Dominio Web	31
2.3.15. Firebase.....	32
3. Capítulo III. Desarrollo del proyecto	33
3.1. Planteamiento del sistema de pánico	33
3.2. Diseño de la aplicación	35
3.2.1. Casos de uso	35
3.2.1.1. Diagrama de Casos de Uso General	35
3.2.1.2. Diagrama de Casos de Uso Específico.....	36
3.2.1.3. Requerimientos Funcionales y No Funcionales	40
3.3. Desarrollo de la aplicación	41
3.3.1. Especificaciones	41
3.4. Implementación de la aplicación	42
3.4.1. Backend	42
3.4.1.1. Firebase	43
3.4.1.2. Servidor Web	44
3.4.2. Frontend.....	47
3.4.2.1. Interfaz de inicio	47
3.4.2.2. Interfaz de comunicación	48
3.4.2.3. Interfaz de Grabar Video	49
3.5. Funcionamiento de la aplicación.....	50
4. Capítulo IV. Análisis de resultados y pruebas	56
4.1. Requerimientos funcionales.....	56
4.2. Requerimientos no funcionales	65
5. CONCLUSIONES Y RECOMENDACIONES.....	66
5.1. Conclusiones	66
5.2. Recomendaciones.....	67
REFERENCIAS	69
ANEXOS	72

1. Capítulo I. Introducción

1.1. Antecedentes

Hoy en día la inversión que realizan las empresas en los sistemas de vigilancia se ha incrementado notablemente. En la actualidad las organizaciones acumulan cada vez objetos de gran valor debido a ello se gastan ingentes cantidades de dinero en su protección con sistemas de vigilancia que contribuyan a su seguridad. No se conforman con contratar guardias de seguridad, que vigile las pantallas de las cámaras de seguridad, sino que ahora se instala sensores térmicos y de movimiento. Es necesario señalar que los sistemas de vigilancia no sólo se usan para cuidar objetos valiosos de las empresas, sino también para vigilancia de entornos, como parques nacionales en lo que se detecta a los cazadores furtivos, etc. Esto demuestra que estos sistemas de vigilancia se usan con mucha frecuencia en la sociedad, lo que conduce a una pregunta: ¿Es posible mejorar la efectividad de estos sistemas? La respuesta es sí es posible mejorar.

Con un sistema automatizado se reduce de manera drástica los problemas de seguridad y se incrementa la efectividad del sistema de vigilancia. Una forma de automatizar el sistema es a través de la vigilancia inteligente, que se refiere a la aplicación de algoritmos, técnicas y métodos de Inteligencia Artificial (IA), con el propósito de desarrollar sistemas de seguridad de vanguardia que ejecuten las tareas de vigilancia y monitorización que por lo general lo realizan los individuos.

Es así, que la automatización de los sistemas de vigilancia inteligente implica una mejora sustancial en la seguridad de las organizaciones, sino también en la seguridad de los individuos; si a ello se le suma un sistema de alerta mediante un botón físico de un teléfono inteligente la seguridad alcanza una sinergia. Cabe destacar que la automatización complementada a un sistema de botón físico mediante un botón físico se vuelve un reto extraordinario al interior de la ciencia, así como de los desarrolladores de sistemas.

El simple hecho de la automatización crea un conflicto de comunicación y coordinación si a ello se suma un sistema de botón físico para alertar una emergencia se vuelve un desafío de creación para el desarrollador de sistemas; puesto que el botón de alerta físico “es una aplicación de seguridad para smartphones que funciona bajo el sistema operativo Android, la cual permite por un lado, el envío de alertas geolocalizadas vía SMS a un grupo de contactos de confianza predefinidos, y por otro lado, la misma alerta geolocalizada podrá ser reportada vía Internet y SMS ante la Central de Monitoreo AsT” (AsT, 2018). En la práctica, “una vez configurada la aplicación, el usuario solamente deberá presionar el botón y el sistema automáticamente enviará las alertas a los contactos preconfigurados, indicando su posición y tipo de alerta” (AsT, 2018).

Uno de los grupos vulnerables en nuestro país corresponde a “los adultos mayores que son personas que tienen 65 años de edad o más; para el 2017 se estima que serán 1.180.944 lo que representará el 7,0% de la población total del país” (INEC, 2017). “El proceso de envejecimiento del adulto mayor difiere de acuerdo a las condiciones sociales, educativas, culturales y económicas que tenga” (INEC, 2017). Dentro del contexto del grupo de personas mayores se encuentran aquellas, que se encuentran en jubilación y que reciben una pensión lo que les permite mantenerse con sus vidas de forma digna por lo que tienen su forma de mantener sus ingresos, también prestados auxilio por parte de sus familiares más cercanos, además otros de estos adultos mayores mantienen sus negocios propios y que se encuentran también actualizados en lo que se refiere a sus estudios tecnológicos.

Otro grupo vulnerable, son las personas con discapacidad que en nuestro país son “449.169” (MINISTERIO DE SALUD PUBLICA, 2018) que se encuentran registradas en el MSP, estas personas también pueden requerir de ayuda ante cualquier emergencia a fin de salvaguardar su integridad, es por ello que se requiere la implementación de una aplicación móvil para alertar una emergencia a través de un botón de alerta instalado en sus teléfonos inteligentes.

1.2. Alcance

Diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables mediante un teléfono inteligente va ser desarrollada en la plataforma Android teniendo en consideración requerimientos tanto de entrada y salida para el ECU911, esto por un lado y por otra parte esta plataforma se enganchará a un servidor que va a permitir notificar a una tercera persona que tenga relación con la persona vulnerable que necesita ser auxiliado; el objetivo básico es enviar una potente alerta a través de un botón físico en un teléfono inteligente.

Mediante el dispositivo de auxilio la persona adulta mayor podrá solicitar ayudar a sus familiares cuando se produzca alguna eventualidad o accidente lo que ponga en serio peligro su vida o salud para lo cual se enviará de forma inmediata un SMS o también un correo con un mensaje donde se determina la ubicación dentro de Google Maps, de esta forma se enviará una respuesta temprana al ECU911.

1.3. Justificación

El botón de alerta, no es otra cosa que “las alarmas anti-coacción y las alarmas anti-atraco que están diseñadas para permitir a los usuarios activar el sistema bajo situaciones específicas de emergencia” (Tyco, 2013). Es necesario indicar que estas alarmas silenciosas son sistemas que no suenan en los espacios que están protegidos, pero avisarán a la distancia al ECU911 de que se produce una emergencia. Este tipo de alarmas por lo general alcanzan una mayor respuesta, debido a la alta posibilidad de que esté teniendo lugar un hecho delictivo, pero a la vez, pueden ser inútiles e incluso fatales si no se utilizan adecuadamente. Por lo tanto, “la capacitación de los empleados para su uso y hacer que practiquen con este tipo de alarmas es de vital importancia” (Tyco, 2013), especialmente en puestos en los que existe una alta rotación de empleados, es esencial.

En este contexto el Servicio Integrado de Seguridad ECU911 de la ciudad de Quito no cuenta con un sistema de botón de alerta inmediata sobre emergencia a personas de la tercera edad que son vulnerables a delitos, esta situación propicia la idea del diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables mediante un teléfono inteligente, en la que habrá la activación intencionada de un dispositivo diseñado para alertar cuando haya una amenaza a la integridad de una persona de la tercera edad. El propósito de este sistema de alerta es permitir que un individuo bajo coacción pueda solicitar asistencia al ECU911 de manera rápida y silenciosa. Cabe señalar que el botón de alerta se utilizará cuando es arriesgado o inseguro solicitar ayuda de otra manera.

Con el diseño e implementación de un sistema de ‘Botón de Alerta’ se aporta una nueva herramienta para alertar con un botón físico de los peligros que podría correr las personas de la tercera edad ante el avance de la delincuencia nacional e internacional que día a día se incrementa y que no ha sido capaz de ser detenida por las instituciones del Estado llamadas a dar protección. El “Botón de Alerta” proporcionará a miles de ancianos en estado de vulnerabilidad una alternativa para evitar un delito y una nueva herramienta en su lucha por alcanzar la seguridad personal.

Se puede estimar que las personas adultas de la tercera edad constituirán los principales beneficiarios tanto del auxilio por parte de este dispositivo, el cual alertará de forma inmediata para que sean atendidos ya sea por robo o por emergencia en la salud de los mismos, todo ello mediante un teléfono inteligente.

El aporte para la Universidad será la propiedad intelectual del desarrollo de la aplicación móvil bajo el sistema operativo Android a la vez que se cumple con un requisito previo para la graduación como profesional.

1.4. Objetivo General

Desarrollar una aplicación móvil bajo el sistema operativo Android que permita detectar un botón físico de un teléfono inteligente para alertar una emergencia de una persona vulnerable.

1.5. Objetivos Específicos

- Determinar las estadísticas del grupo de personas vulnerables que existen en la ciudad de Quito.
- Diseñar una aplicación móvil para teléfonos inteligentes que permita alertar una emergencia de personas vulnerables a través de una interfaz (botón integrado) del smartphone.
- Establecer una comunicación entre la aplicación móvil y el servidor de base de datos para la visualización y actualización instantánea de registros de detección del uso de un botón integrado del teléfono inteligente.
- Realizar pruebas de la aplicación, a través de las técnicas de prototipado.

1.6. Metodología

En referencia a la metodología se trabajará tomando el aplicativo móvil bajo el sistema operativo Android el cual se desarrollará en el modelo en cascada creado por Pablo Domínguez, el cual consiste en un desarrollo secuencial para llegar a la aplicación móvil que se forma mediante un conjunto de etapas que se desarrollan una después de otra. Toma este nombre por las diferentes etapas que conforman la aplicación, las cuales van colocadas una detrás de otra y siguiendo un camino que va de arriba hacia abajo.

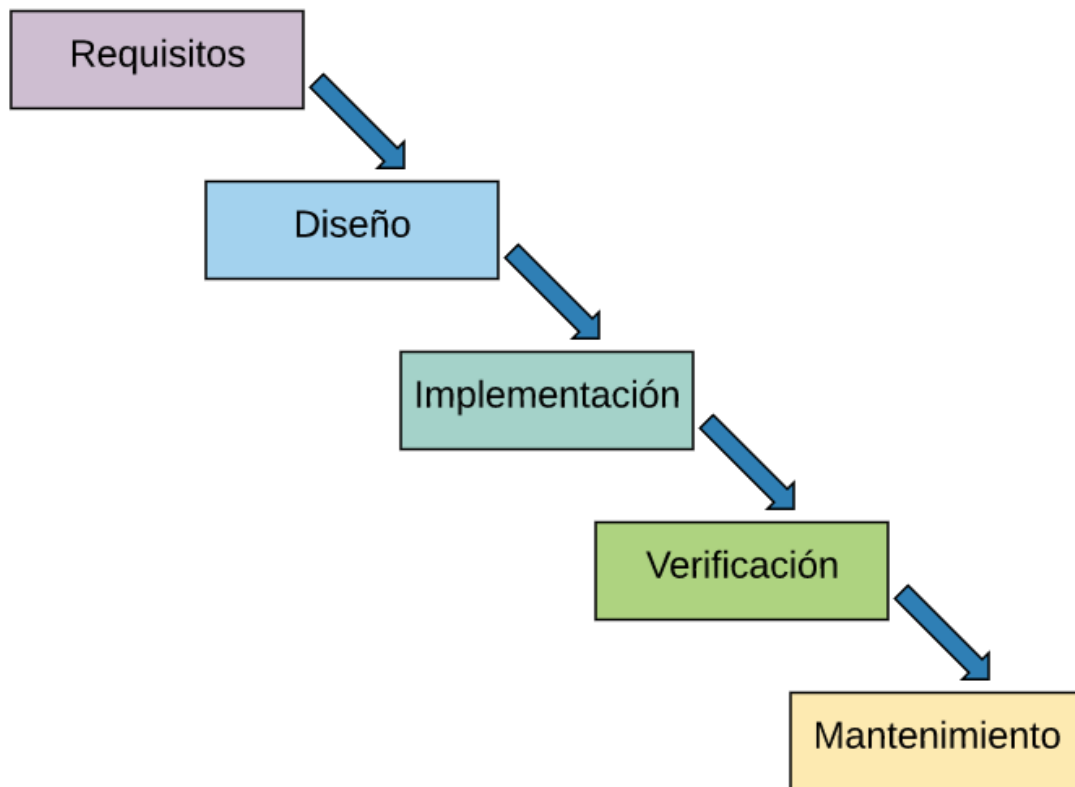


Figura 1. Las etapas del modelo de cascada.

Tomada de (Domínguez, 2017).

1.6.1. Fases del modelo

De acuerdo al modelo este sigue diversas fases en escala y de una manera secuencial, donde la etapa siguiente empieza cuando termina la etapa anterior.

- **Etapa 1: Requisitos del software.** - Se hace un análisis de las necesidades de una persona vulnerable para alertar una emergencia a través de un botón físico para determinar las características de una aplicación móvil con el sistema operativo Android, y se especifica toda la aplicación móvil sin ahondar en detalles técnicos.
- **Etapa 2: Diseño.** - Se describe la estructura interna de la aplicación móvil, y las relaciones entre las entidades que lo componen como el ECU911. Es decir, el sistema se conforma por elementos que puedan elaborarse por separado de manera secuencial.

- **Etapa 3: Implementación.** - Es aquí en lo cual se realiza una verdadera programación de lo fundamental en lo referente a los requisitos de acuerdo a lo que se estima en la estructura anterior, con ello se llega a la formulación del problema y después a un programa el cual estará a cargo de la ejecución de los algoritmos útiles para resolver el problema planteado.
- **Etapa 4: Verificación.** - Se verifica que todos los componentes del sistema funcionen correctamente y cumplen con los requisitos que se necesita. El objetivo de hacer pruebas es disponer de información de calidad para: defectos o bugs, aumentar la calidad de la aplicación, refinar el código escrito sin miedo que este llegue a ser roto o introducir nuevos bugs, etc.
- **Etapa 5: Instalación y mantenimiento.** - Al terminarse el desarrollo de las funcionalidades del software y haber comprobado que funcione, se realiza la fase de instalación y también el mantenimiento, donde se instalará la aplicación móvil y se comprueba su entorno de uso, en lo posterior se asegurará que funcione y también el mantenimiento al mismo.

1.6.2. Los métodos de investigación

Los métodos escogidos para esta investigación serán:

- **Método analítico.** - Se distinguen los elementos de la aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente y de inmediato se revisa de forma ordenada uno a uno de forma individual.
- **Método sintético.** - Se trata de un proceso de desarticulación mental de la aplicación móvil en sus partes y de reunificación del todo a base de sus partes.

- **Método inductivo.** - Es el razonamiento lógico e intelectual que, partiendo del diseño e implementación de una aplicación móvil, se eleva a conocimientos generales del desarrollo de una aplicación móvil bajo el sistema operativo Android. Este método también permite la formación de hipótesis.
- **Método deductivo.** - Mediante el método lógico deductivo se destinan los elementos observados para el desarrollo de aplicaciones móviles, a partir de un enlace de juicios al diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente.

2. Capítulo II. Marco Teórico

En esta sección se expondrán datos e información de las bases teóricas – conceptuales que sirven de fundamento para el desarrollo e implementación de una aplicación móvil en la plataforma Android con requerimientos de entrada y salida a un centro de atención de emergencias.

2.1. Estadísticas de discapacidades

2.1.1. Personas con discapacidad registradas

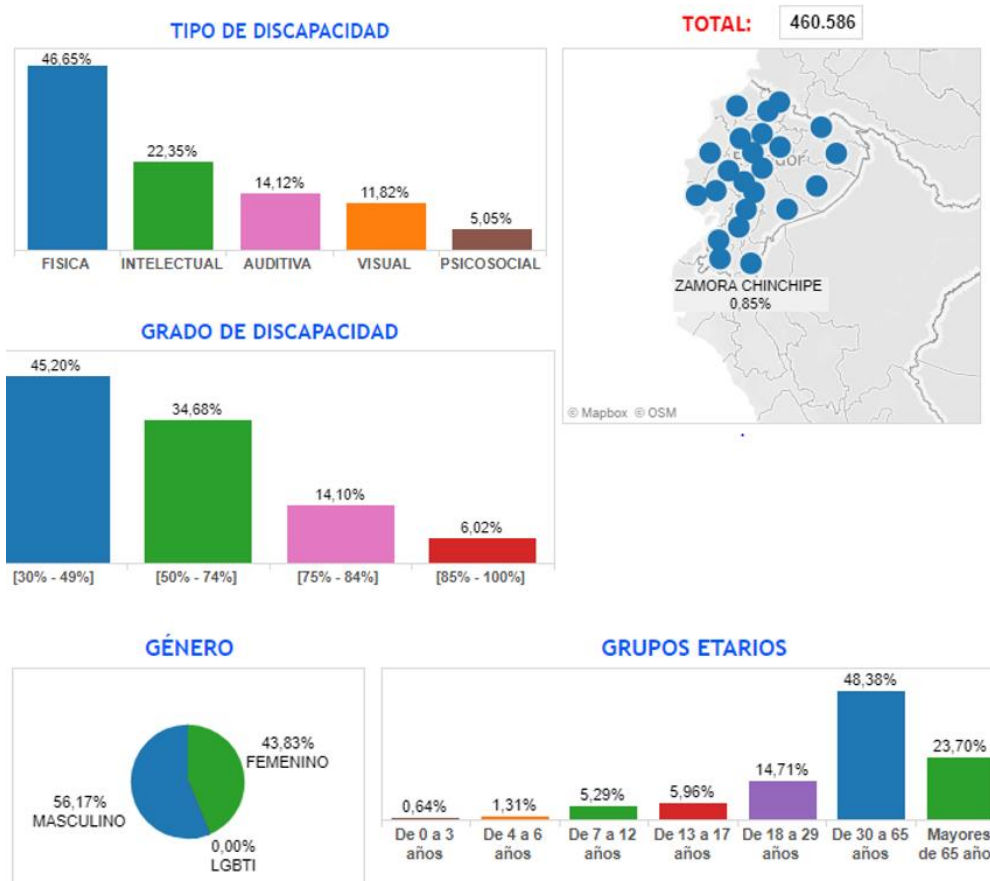


Figura 2. Personas con discapacidad registradas.

Tomada de (MSP, 2019).

2.2. Situación sociodemográfica del adulto mayor

2.2.1. Población Adulto Mayor

Con el propósito de conocer la situación de la población adulto mayor, es fundamental el incluir algunos datos que se tiene en base a una población total de la ciudad de Quito. Por lo que en la tabla 1 producto de la Encuesta Condiciones de Vida (ECV) 5ta. Ronda, la población total de la ciudad de Quito es de **1.523.274** habitantes. Esta población está formada por **746.923 (49,0%)** hombres y **776.352 (51,0%)** mujeres (Morales, 2008).

Tabla 1.

Quito, Población resultante, por edad, según sexo.

Grupo de Edad	SEXO		Total
	Hombre	Mujer	
0 a 4	91.856	66.941	158.797
5 a 9	70.911	73.192	144.102
10 a 14	68.178	75.823	144.001
15 a 19	67.472	68.974	136.446
20 a 24	81.498	78.693	160.191
25 a 29	65.887	68.459	134.346
30 a 34	57.680	56.827	114.507
35 a 39	54.606	53.482	108.088
40 a 44	39.369	55.710	95.079
45 a 49	33.973	37.849	71.822
50 a 54	35.838	43.873	79.711
55 a 59	24.231	30.468	54.699
60 a 64	14.458	16.400	30.858
65 a 69	14.849	16.748	31.597
70 a 74	10.926	8.804	19.730
75 a 79	10.520	11.842	22.363
80 a 84	2.026	7.152	9.178
85 a 89	770	3.654	4.424
90 a 94	1.466	1.460	2.926
95 y más	409		409
Total hab.	746.923	776.352	1.523.274

Tomada de (INEC, 2008).

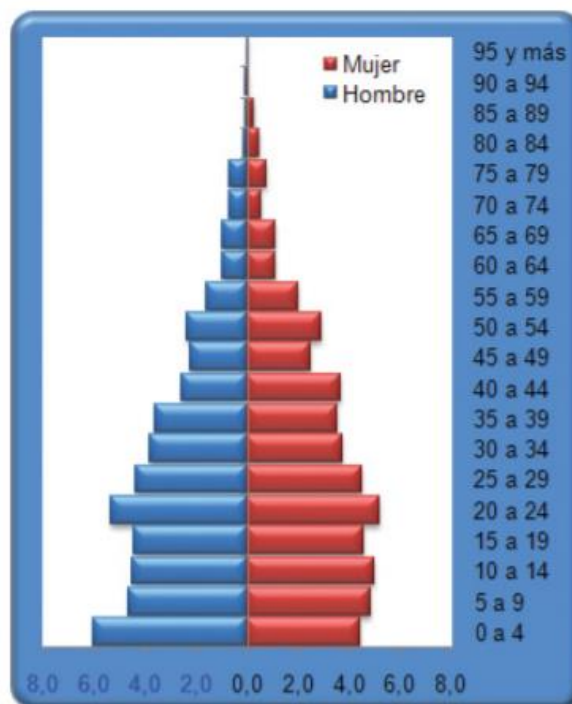


Figura 3. Quito, Pirámide de la población total, por grupo de edad, según sexo. Tomada de (INEC, 2008).

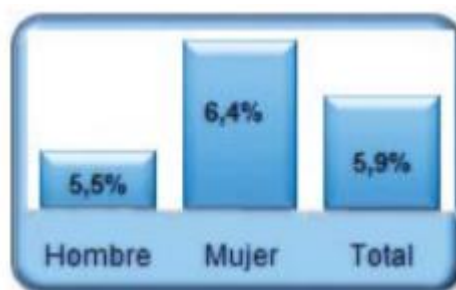


Figura 4. Quito, Porcentaje de población adulto mayor, por sexo. Tomada de (INEC, 2008).

De acuerdo a los datos del INEC se puede determinar que los adultos mayores que se encuentran en la ciudad de Quito constituyen de ambos sexos un total de 90.628 (5,9%) en referencia a la población total de esta ciudad. Esto también se da para adultos mayores hombres 40.967 (5,5%) y también en adultas mayores mujeres 49.660 (6,4%).

2.3. Fundamentación teórica

2.3.1. Sistemas de Información

Para los autores Laudon un sistema de información puede denominarse como un conjunto de elementos interrelacionados que se encargan de recopilar, procesar, almacenar y distribuir información que sirva para dar soporte a los procesos para la toma de decisiones, así como el control en una institución. Además, “de dar soporte a la toma de decisiones, la coordinación y el control, los sistemas de información contribuyen a apoyar al nivel ejecutivo y operativo acerca del conocimiento a analizar problemas, visualizar temas complejos y crear nuevos productos” (Laudon & Laudon, 2012).

2.3.2. Base de datos

Una base de datos “se puede conceptualizar como una colección por lo general inmensa de datos organizados particularmente para la búsqueda, así como para la recuperación rápida”. (Merriam-Webster, s.f.).

Por lo general, cuando se refiere a una base de datos en realidad se refiere al modo de gestión de base de datos (DBMS), lo que consiste en un programa que se encarga del acceso y almacenamiento de los datos. Se puede citar como ejemplo: de DBMS los que incluyen Oracle, Microsoft SQL Server, MySQL, MongoDB, PostgreSQL, Sybase, MariaDB y DB2.

En cuanto a los tipos de bases de datos, existen algunos que se pueden enunciar: No obstante, hoy en día el modelo más utilizado es el relacional (DB-Engines Ranking, 2017), que se halla asociado al lenguaje SQL, aunque para algunos casos nada más, en los últimos 10 años se ha vuelto popular “las denominadas bases de datos NoSQL, que son llamadas como no relacionales (NoSQL definition), en función a las necesidades de bases de datos más ágiles para aplicaciones en tiempo real así como almacenaje de documentos en gran volumen”.



Figura 5. Comparación simple SQL y NoSQL.

Tomada de (Vaas, 2016)

2.3.3. Arquitectura Cliente – Servidor

En lo inherente a la arquitectura cliente/servidor, ésta comparte tareas de procesamiento de datos en un servidor, por lo general, se da en un escritorio de trabajo de alto nivel, pero es posible que sea un mainframe, y los clientes que utilizan las PC. Las computadoras de escritorio poseen una gran capacidad de procesamiento de datos, por lo expuesto, son capaces de tomar información sin procesar, devueltos por el servidor, así como formatear el resultado, producto de salida. Las aplicaciones, así como los procesadores de consultas se pueden ejecutar y almacenar en las PC. El tráfico que se produce en la web consiste en las solicitudes de manipulación de datos enviadas desde la computadora al servidor que contiene la base de datos; no obstante, los datos sin procesar se restituyen como resultado de esa solicitud. “El resultado

consiste en un tráfico de red significativa menor, así como un rendimiento en teoría mejor” (Harrington, 2016).

2.3.4. Lenguaje de programación

Consiste en un lenguaje formal inherente a un conjunto de instrucciones que se han diseñado para crear programas nuevos para computadoras.

2.3.5. Aplicaciones móviles

Se denomina a una aplicación móvil que contiene un programa de PC diseñado para ejecutarse en un dispositivo móvil que generalmente se da en un Smartphone o una Tablet, no obstante, su uso, con las transformaciones del caso, se ha ampliado a otros dispositivos como relojes inteligentes, centros de entretenimiento de televisores y automóviles. En el caso propuesto se va concentrar en los teléfonos inteligentes (Smartphone).

El avance de aplicaciones móviles tiene en cuenta las restricciones, así como las características de cada uno de estos dispositivos, por lo general, “su funcionamiento se produce con batería y poseen procesadores que son menos potentes que las computadoras de escritorio, pero a la vez poseen funciones innovadoras como geo-localización, pantallas táctiles y cámara fotográfica de altos megapíxeles”.

El componente fundamental en lo que se refiere a las aplicaciones móviles en la época actual; se puede indicar que es el diseño de la interfaz del usuario, la cual deberá ser de fácil uso, amigable y también ser intuitiva, y también es necesario que presente un adecuado funcionamiento siempre respetando los factores y limitantes.

2.3.6. Sistemas Operativos móviles

Esta palabra, tiene que ver con los sistemas operativos que son para teléfonos móviles, tablet y relojes inteligentes, dispositivos como como laptop que se consideran móviles, utilizan sistemas operativos concebidos de manera inicial para PC de escritorio por lo que no ingresan en esta categoría de producto.

2.3.7. Sistemas Operativo Android

Según lo que manifiesta el autor Andy Rubín acerca del sistema operativo “Android es una distribución de Linux que fue diseñado para ayudar a los clientes a crear diferentes aplicaciones, a la vez que se aprovecha varias particularidades y utilidades de sus teléfonos”.

El sistema operativo Android se trata de un sistema operativo para dispositivos electrónicos que son móviles, este sistema fue creado por la empresa de Internet Google, el cual se fundamenta en otros sistemas operativos, como GNU/Linux.

En otras palabras, se trata de una plataforma que posee un código abierto, que cualquier individuo en la Tierra tiene la facultad de editar su programación a la vez que permite el desarrollo de varias aplicaciones, para ello se debe contribuir a crear herramientas de desarrollo, depuración, compilación y emulación, para verificar la efectividad del software que se desea diseñar.

A través de la investigación efectuada se da a conocer las diversas definiciones del sistema operativo Android; además, se explica su nivel de utilización de manera local e internacional, constituyéndose en uno de los sistemas más utilizados en Sudamérica por tratarse de un sistema operativo libre para el usuario.

2.3.8. Características de Android

En lo siguiente, se indica un resumen donde se presentan las características más fundamentales de Android:

Tabla 2.

Características de Android.

CARACTERÍSTICAS	DESCRIPCIÓN
Diseño para dispositivos pequeños	Este sistema operativo tiene compatibilidad con pantallas VGA (y mayores, ¡llega ya hasta los 4K!), posee gráficos 2D y 3D, que ya está presente en la

	mayoría de teléfonos inteligentes.
Almacenamiento	Dentro de lo que es el almacenamiento presenta el SQLite, donde se guardan los datos que llegan de las aplicaciones.
Conectividad	El sistema Android se encuentra adaptado a lo siguiente en conectividad: CDMA, UMTS, Bluetooth, GSM/EDGE.EV-DO, es fundamental indicar que este tipo de aplicativos se hallan ya en teléfonos actuales.
Mensajería	En referencia a la mensajería se determina que esta está dentro de Android Cloud to Device Messaging Framework (C2DM), lo que viene a ser parte del servicio que brinda Push Messaging de Android en o referente a MSM.
Navegador web	El navegador web que viene en Android se basa en el motor del navegador que posee un código abierto WebKit. Se debe señalar que el navegador es muy versátil y eficiente porque ayuda a cargar páginas web en forma rápida. Se fundamenta en el motor JavaScript V8 de Google Chrome.
Soporte de Java	En la época actual la mayoría de las aplicaciones se encuentra realizadas en el lenguaje Java lo que determina el aplicar el código en el sistema operativo. Este tipo de código se encuentra en un ejecutable Dalvik la cual se ejecuta dentro de Dalvik que constituye una máquina de manera especializada en lo referente a sistemas virtuales y que opera en Android con batería los cuales tiene memoria, así como un procesador de forma restringida. Cuando se halla la versión 5.0 de Android es utilizada con una nueva máquina virtual que se la llama ART (Android Runtime) la que permite realizar una verdadera compilación antes de tiempo.

Soporte multimedia	El sistema Android se halla configurada para formatos como son: 3GP o MP4, MIDI, WAV,PNG,GIF Y BMP también para los formatos H.263, H.264, AAC,SP, AMR.
Soporte para streaming	Android también se encuentra configurada para streaming: RTP/RTSP, donde se desarrolla una descarga progresiva de HTML (tag <video> de HTML5). Adobe Flash Streaming (RTMP), que determina la instalación de Adobe Flash Player, solo para varios terminales. En la actualidad Flash Player ya no es útil para Android y ha sido transformado para HTML5.
Soporte para hardware adicional	Android se caracteriza por disponer cámaras de video, pantallas táctiles, acelerómetros, GPS, giroscopios, sensores de proximidad, magnetómetros, y de presión, termómetro, presión sanguínea, lector de huellas dactilares, aceleración 2D y 3D, etc.
Entorno de desarrollo	El desarrollo en el ambiente de Android Studio mediante un emulador de dispositivos, el cual constituye un instrumento de saneamiento y examen de rendimiento. Tradicionalmente se usaba Eclipse y el Plugin que se trata de instrumentos de desarrollo de Android (ADT), sin embargo con el paso del tiempo este se volvió obsoleto y ya no cuenta con soporte de Google. No es posible recomendar a los estudiantes usar este entorno; puesto que no permite la instalación de versiones últimas para Android de forma progresiva y completa.
Mercado de aplicaciones	Google Play se trata de un catálogo de aplicaciones que son gratuitas así como también de pago que es posible bajarlas e instalar en los definitivos dispositivos Android.

Multi-táctil	Android presenta un soporte origen para pantallas multi-táctiles que permiten el navegar en la pantalla táctil con varios dedos.
Bluetooth	En la versión actualizada 2.2 para Android, se ha incluido una funcionalidad completa.
Videollamada	Android incluye la video-llamada como posibilidad para el usuario mediante de Google Talk.
Multitarea	Presenta la opción de multitarea real de aplicaciones, dicho de otra forma, las aplicaciones que no rinden en primer plano tienen ciclos de reloj del procesador que facilitan el poner al día su estado.
Características basadas en voz	Se cuenta con la posibilidad de dar órdenes de voz al terminal. Así en el siguiente caso, la búsqueda en Google mediante una orden de voz; esta se encuentra desde la primera versión del sistema operativo de Android.
Tethering	Android permite compartir la conexión a Internet (en inglés, tethering), que facilita el utilizar el teléfono inteligente el cual se sitúa como punto de acceso inalámbrico, en referencia al ordenador donde se puede utilizar la conexión 3G o 4G del móvil para Android.

Tomada de (Robledo, 2017).

2.3.9. Arquitectura de Android

Se especifican los fundamentales elementos que tiene la arquitectura del sistema operativo Android los cuales son:

- **Aplicaciones:** El lenguaje de programación Java está escrita en todas las aplicaciones. Estas apps están incluidas para uso del usuario y estas son: correo electrónico, programa de SMS, calendario, mapas, navegador, contactos, etc.

Cabe señalar que todas estas aplicaciones en sistema operativo Android manejan y operan los servicios y sistemas:

- Un grupo de conjunto (**Views**), los cuales facilitan crear las interfaces de usuarios, como por ejemplo las tablas, las listas y cajas de texto.
 - Proveedores de contenidos (**Content Providers**), permiten a las aplicaciones el acceder a la mayor parte de información donde se encuentran las aplicaciones: por lo que se puede mencionarlos a los contactos del teléfono y a la compartición de la mayoría de datos.
 - Receptor de mensajes (**Broadcast Receiver**), se refiere a los anuncios broadcast del sistema. Por ejemplo, se utiliza para que una aplicación recepte determinados mensajes del sistema operativo como es la pantalla ya sea en modo apagado. Además, se puede diseñar mensajes propios.
 - Gestor de recursos (**Resource Manager**), que permite el acceder a las posiciones o recursos los cuales no llegan desde el código fuente, como en las imágenes, los textos de internacionalizaciones y también los ficheros de los estilos como son los de layout.
 - Gestor de notificaciones (**Notification Manager**), mediante esta aplicación se puede indicar de forma instantánea alertas que se indican en la barra de estado de Android.
 - Gestor de actividades (**Activity Manager**), donde se puede analizar el periodo de vida de cada aplicación.
 - Servicios (**Services**), elemento de software que se ejecuta en segundo plano y efectúa tareas repetitivas como sincronizaciones de datos en Internet. No cuenta con una interfaz gráfica.
- **Marco de desarrollo de aplicaciones:** Cada programador tiene paso amplio a las diversas aplicaciones (bibliotecas) del Framework (marco de desarrollo), utilizado por las aplicaciones que constituyen la línea base. En lo que se refiere a la arquitectura está diseñada para simplificar la reutilización de elementos, dicho de otra manera, cualquier aplicación permite visualizar sus capacidades, así como hacer uso de tales capacidades.

- **Bibliotecas nativas:** El sistema operativo Android incluye un conjunto de bibliotecas de C/C++ usadas por algunos elementos del sistema. Después de cada esquema es posible visualizar la arquitectura del sistema operativo Android en la cual hay una descripción de tales bibliotecas.
- **Runtime (ejecutable) de Android:** El sistema operativo Android, además incluye un grupo de bibliotecas que determinan la línea base que ofrecen en su mayor parte las funciones del lenguaje de programación Java. Por lo cual cada aplicación del sistema operativo Android ejecuta unas actividades de forma individual en el entorno de la máquina virtual Dalvik que ejecuta ficheros del tipo.dex (Dalvik Executable), que se trata de un formato optimizado que contribuye a ahorrar memoria. La versión del sistema operativo Android 5.0 fue reemplazada por Dalvik por ART. Esta nueva máquina virtual producto del mejoramiento continuo, se encarga de reducir el tiempo de ejecución del código Java hasta en un 33%.
- **Núcleo Linux:** El sistema operativo Android se fundamenta en Linux en lo que se refiere a los servicios que constituye la base de su sistema, como son: controladores, seguridad de procesos y también la gestión de memoria.

A continuación, en la figura 6 es posible apreciar el diagrama de la arquitectura del sistema operativo Android:

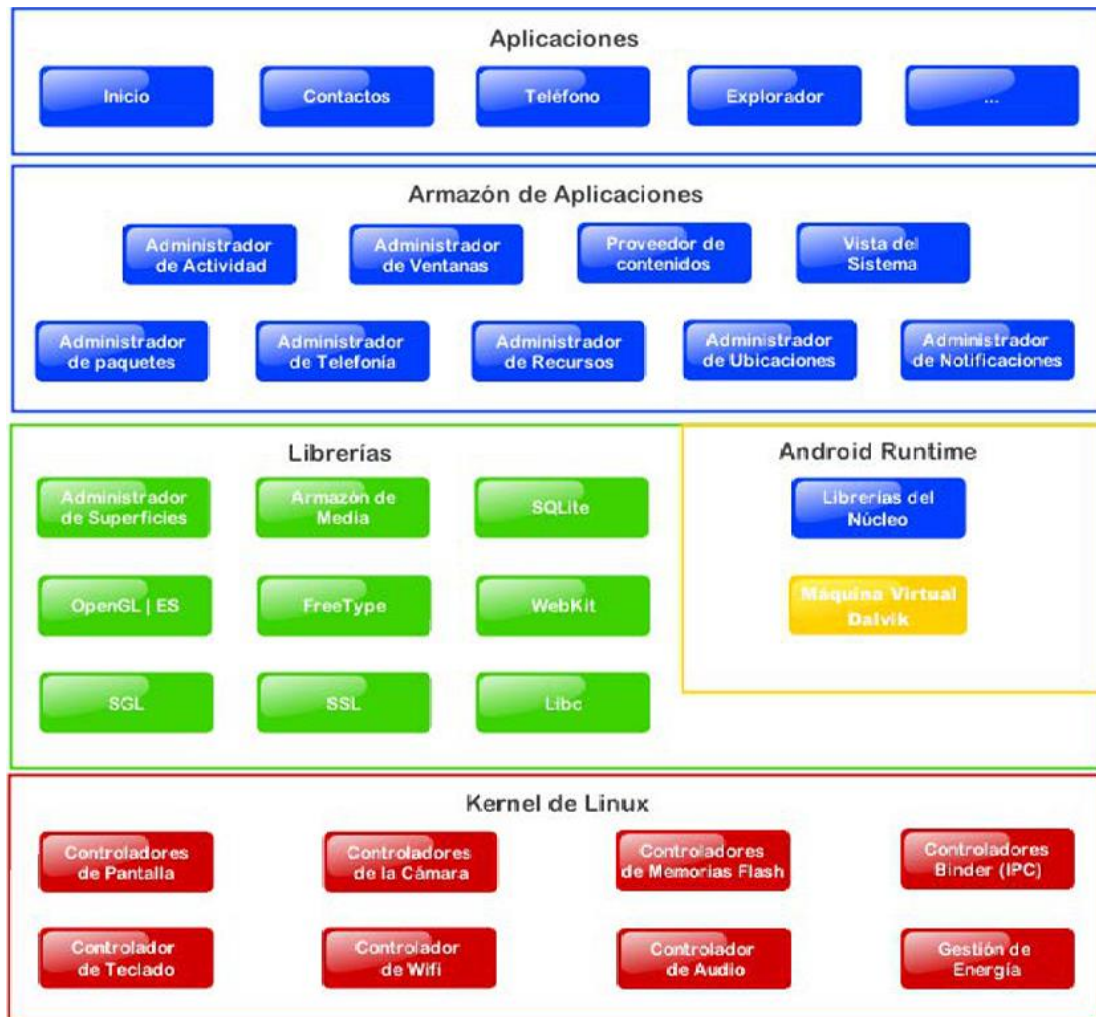


Figura 6. Diagrama de la arquitectura de Android.

Tomada de (Robledo, 2017).

En la figura 6, en color verde, se encuentra un conjunto de bibliotecas en C/C++ utilizadas por algunos elementos del sistema operativo Android, las mismas que se encuentran compiladas en código nativo del procesador. La mayoría de estas bibliotecas se fundamentan en proyectos de código abierto. A continuación, se detallan estas bibliotecas:

Libc: Proviene de una derivación de la librería BSD de C estándar (libc), que sirve de soporte para dispositivos que se fundamentan en Linux. Android Studio en su versión 1.3 ayuda a utilizar de forma directa como lenguaje de programación C en lugar de Java.

Armazón de Media (Media Framework): Se trata de una biblioteca que se fundamenta en PacketVideo's OpenCORE que soporta codecs para grabación y reproducción de variedad de formatos de audio vídeo e imágenes MP3, MPEG4, AAC, H.264, AMR, JPG y PNG.

Administrador de superficies (Surface Manager): Se encarga de administrar el acceso al subsistema de representación gráfica en 2D y 3D (OpenGL).

WebKit: Soporta un avanzado navegador web usado en Android y en la vista webview. Consiste en la misma biblioteca que utiliza Google Chrome y Safari de Apple.

SGL: Motor de gráficos 2D.

OPENGL: Se halla fundamentada en OpenGL ES 3.1. Esta biblioteca usa hardware 3D (acelerador), del dispositivo si éste está disponible o el software para el dibujo en 3D.

FreeType: Se refiere al tipo de fuentes en bitmap así como de renderizado vectorial.

SQLite: Es aquel potente y ligero motor de bases de datos relacionales que se halla disponible para cualquiera de las aplicaciones.

SSL: Se encarga de ofrecer un conjunto de servicios de cifrado Secure Socket Layer (capa de conexión segura).

2.3.10. Android Studio

Es un medio ambiente que contribuye al desarrollo de aplicaciones móviles para Android Studio, éste consiste en un editor de códigos y de un innumerable número de herramientas, así como de funciones que incrementan la productividad durante la operación de las aplicaciones móviles y ofrece las características siguientes:

- Compilación Flexible.
- Emulador de alta velocidad con algunas funciones.
- Medio ambiente unificado para el desarrollo de aplicaciones en cualquiera de los dispositivos Android.
- Es posible aplicar modificaciones mientras la aplicación se encuentra en ejecución, sin la necesidad de una nueva compilación.
- Integra plantillas de códigos para un incremento de efectividad de compilación de funciones.
- Una holística gama de herramientas y frameworks.
- Compatibilidad con C++ y NDK.

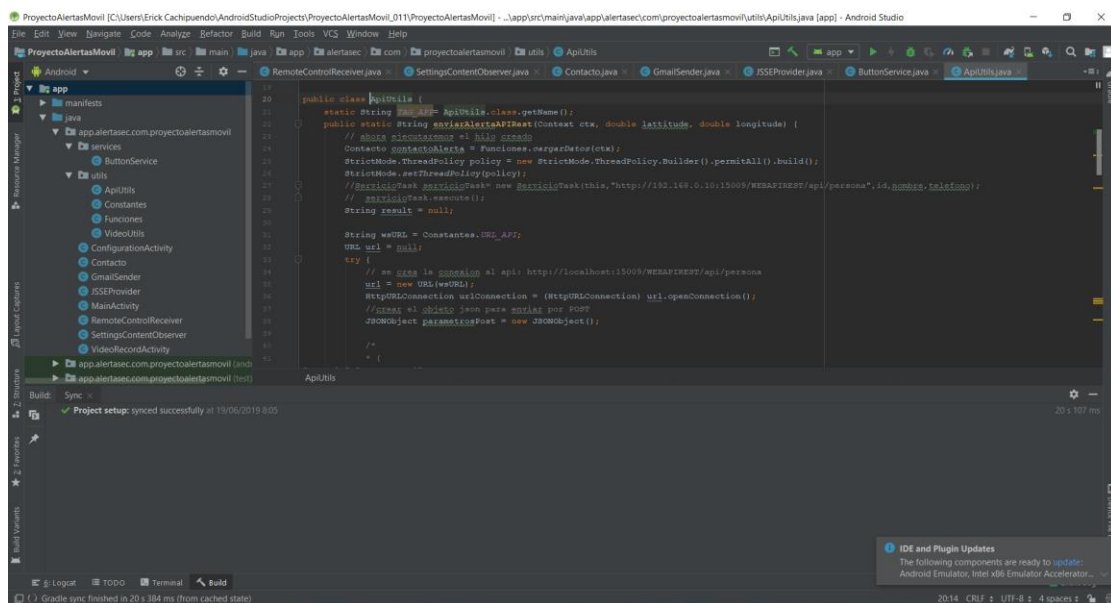


Figura 7. Interfaz Android Studio.

Se puede indicar que las aplicaciones que se dan para el programa Android se conforman con uno o también con varios elementos siendo los siguientes: actividades, servicios también receptores de los mensajes, así como de proveedores de los contenidos. Se puede indicar que estos, cuentan con una determinada funcionalidad en la aplicación, por lo que se puede iniciar cada uno de estos elementos de forma individual, es así que se determina que otras están en poder activarlos. Es el archivo en donde se encuentran todos los elementos el mismo que utiliza la aplicación donde existen los requisitos, mediante los cuales se desarrolla el funcionamiento de la misma, por ejemplo,

configuración mínima de hardware, se puede especificar una versión pequeña de Android, también la seguridad, etc. También se especifica el código el mismo que no constituye de la aplicación, sino que este es de imágenes, así como de las cadenas de internacionalización, además de otros recursos multimedia, se especifica el diseño que se da en multimedia y también el diseño que se tiene en la interfaz del usuario, etc. Es así que se puede agregar variadas configuraciones tomando en cuenta el idioma en el que está el teléfono o también en virtud de la interfaz del cliente, y también se puede cambiar para los diferentes tamaños de pantalla.

Se puede determinar que en lo referente a las aplicaciones que son del sistema operativo Android se encuentran codificadas en lo que es el lenguaje Java. Por lo que estas herramientas tanto de SDK permiten que Android procese y compile el código, en conjunto con la mayoría de datos tomando en cuenta los archivos de recursos, en que estén en un ambiente Android. Se determina que el archivo tiene la extensión apk la cual es de Android que permite se instale la aplicación.

Al ser ejecutado el principio “privilegios mínimos” en el entorno del sistema operativo Android este se ejecutará al inicio, además es conocido que el sistema Android se basa totalmente en el Kernel de Linux.

Es decir, pequeñas partes de la aplicación tiene el acceso a la mayoría de los elementos que son fundamentales en lo referente al desarrollo de la labor. Y en lo referente a la última versión que se tiene del sistema operativo Android, el usuario está en la capacidad de dar por concluida la aplicación de forma inicial. Es por ello que el ambiente es seguro en donde está diseñada la aplicación la cual no está permitida en el acceso a las partes que se encuentran en el sistema si esta no cuenta con el acceso necesario.

2.3.11. Programación sobre Android

El proceso de diseño de aplicaciones donde se desarrollan en el sistema operativo Android facilita de forma general la utilización del lenguaje de

programación Java y Android Studio de forma IDE oficial. Aunque según el autor Kotlin, “un nuevo lenguaje de programación basado en Java también está totalmente soportado” (Rodríguez, 2017).

2.3.12. Componentes de las aplicaciones

2.3.12.1. Actividades

Al analizar cómo funciona una actividad esta tiene una pantalla con la cual se puede interactuar con los usuarios lo que le permite realizar una determinada acción entre las más usuales es marcar un número de teléfono, realizar una foto, enviar un correo electrónico o también ver un mapa. Es por ello que a esta actividad se le determinará una ventana mediante la cual puede realizar el dibujo de la interfaz con el usuario. Se indica también que esta ventana cubrirá toda la pantalla, pero también esta puede ser más reducida y puede quedar flotando encima de otras ventanas.” (Developers, 2019).

Se puede determinar que las aplicaciones constituyen un conjunto de actividades las cuales se encuentran vinculadas entre sí. Se puede evidenciar que una determinada aplicación es aquella que se indica como actividad principal la cual se muestra al usuario cuando se inicia esta por una primera vez. También se puede determinar que cada actividad desarrolla otra actividad de forma secuencial lo que permite el realizar diferentes acciones, es por ello que cuando se da una actividad también se mantiene activa la otra que le sigue y así sucesivamente, es por ello que el sistema permite mantener una pila de estas actividades en tiempo real, es por ello que cuando se inicia una actividad nueva esta se mantiene en la pila, por lo que se capta el interés del usuario.

Se tiene un proceso donde la última actividad en entrar será la primera en salir y así sucesivamente, donde el usuario al terminar de interactuar con lo que se refiere a la actividad que está activa, presiona el botón que retrocede y se quita la cola de actividades por lo que se elimina la actividad y luego sigue la que está en cola o la actividad precedente. (Developers, 2019).

De acuerdo al método callback se puede detener la actividad al ser iniciada otra y con ello se notifica el cambio lo que determina un ciclo de vida de la actividad o lo cierra. Es por ello que se indica, que existen variados métodos callback los cuales reciben este tipo de actividades como el resultado, cuando se da el cambio de estado. (Por cuanto se indica que el sistema se encuentra en proceso de cambio, reanudación, o destrucción), es así que mediante este método se puede realizar la tarea específica la cual es óptima en lo referente a esta modificación de estado. Por ejemplo, cuando se para una actividad la cual permite el liberar los objetos que son amplios como lo es, una conexión de la red con la base de datos. Es por ello que al eliminarse esta actividad también se permite volver a recuperar las acciones que quedaron o que se interrumpieron, es por ello que todas las acciones de transferencia de estado conforman el ciclo de vida de lo que es una actividad. (Developers, 2019).

2.3.12.2. Ciclo de vida

Es de vital importancia la gestión del ciclo de vida de las actividades para diseñar una aplicación sólida y flexible. Esta actividad cuenta con tres estados:

- Activa (Running): Mediante esta opción se posiciona en la parte superior de la pila lo que permite que se la visibilice.
- Visible (Paused): Se la puede ver a la actividad, pero esta no tiene el foco, solo cuando se activa tiene el estado lo que representa que sea transparente o que no esté en toda la pantalla. Cuando pasa a estar detenida es cuando esta está tapada.
- Parada (Stopped): Se especifica este estado cuando la actividad no es visible. El que está desarrollando estará en capacidad de guardar el estado de la interfaz de usuario, así como también las preferencias, etc.
- Destruída (Destroyed): Se produce ello cuando se elimina mediante el método finish(), también cuando se la destruye por el sistema.

Cuando se produce la modificación de una actividad esta permite el iniciar eventos los cuales puede ser atrapado por los diversos métodos de la

actividad. Se puede evidenciar este tipo de métodos mediante el esquema que captura este tipo de eventos.

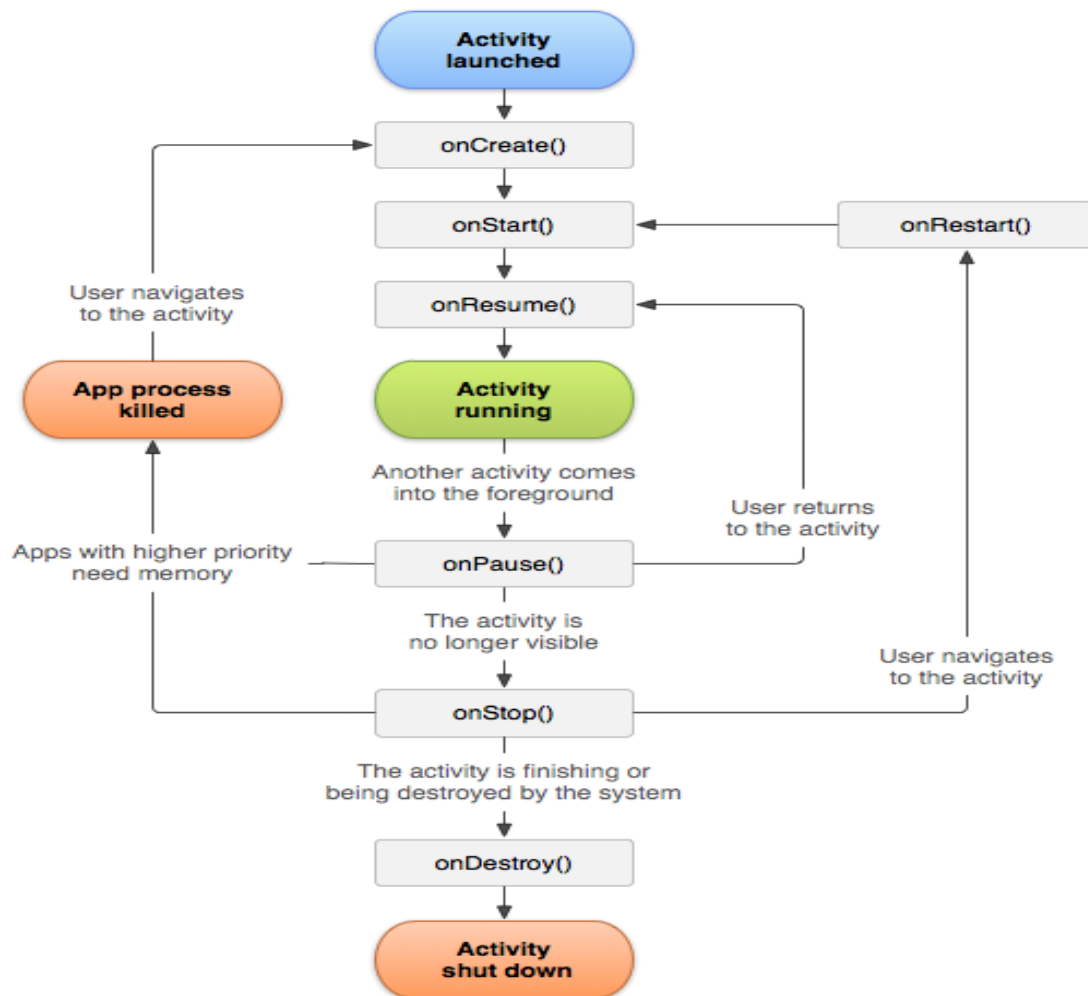


Figura 8. Periodo ciclo de vida de una actividad.

Tomada de (Developers, 2019).

onCreate(Bundle): Se denomina así a la creación de la actividad. Se usa para efectuar todo tipo de inicios, como la creación de la interfaz de usuario o al inicio de estructuras de datos. Puede recibir información de estado de la actividad (en una instancia de la clase Bundle), por si se reinicia desde una actividad que ha sido parada y vuelta a crear.

onStart(): Se refiere a la actividad que está en la cola de ser mostrada al usuario.

onResume(): Se denomina así cuando la actividad va a iniciar e interactuar con el usuario. Se trata de un espacio para lanzar las animaciones y la música.

onPause(): Indica que la actividad está a punto de ser lanzada a segundo plano, normalmente porque otra actividad es lanzada. Es el lugar adecuado para detener animaciones, música o almacenar los datos que estaban en edición.

onStop(): Donde la actividad no se la puede ver por parte del usuario. Es por ello que si no se tiene una excelente retención esta puede llegar a desaparecer o que sea destruida sin dar nombre a este tipo de método.

onRestart(): Este tipo de actividad es aquella la cual se volverá a presentar luego de ser pasada por onStop().

onDestroy(): Este nombre determina que una actividad será destruida casi totalmente. Por ejemplo, cuando se determina que el usuario ha aplastado el botón retroceder o cuando se utiliza el método finish() por lo que si no existe la suficiente retención se puede llegar a destruir sin requerir este método.

2.3.12.3. Diseño de interfaces de usuario en Android

El sistema operativo Android ofrece un marco flexible que se requiere para el diseño de interfaces de usuario, de manera que pueda indicar distintos diagramas gráficos en varios dispositivos, crear elementos y personalizados e incluso modificarles en función de la disposición de la pantalla.

El diseño de una determinada interfaz gráfica de usuario se trata de una perspectiva que se pasa por alto al dar origen a una aplicación, lo cual tiene un impacto en la usabilidad debido a que, en estas aplicaciones, por la naturaleza táctil de su pantalla que existe en los dispositivos móviles, demuestra que hay activa interacción con la interfaz.

El explorador de Google publicó Material Design, “que se ha convertido una guía integral para el diseño visual, de interacción, así como movimiento en diferente plataformas y dispositivos” (Google, s.f).

2.3.12.4. Shared Preferences

Las preferencias compartidas (shared preferences), se trata de una manera que ofrece el sistema operativo Android para guardar datos primitivos como son: flotantes, booleanos, long, enteros y cadenas. Estos datos se conservan en las sesiones de usuario, aunque se cierre la aplicación, para utilizar shared preferences se utiliza uno de los dos métodos siguientes:

getSharedPreferences(): Se usa esta opción cuando se requiere de algunos archivos de preferencias identificadas por un nombre, que se refiere al primer parámetro.

getPreferences(): Se usa esta alternativa si sólo se requiere de un archivo de preferencias para esta actividad. Por lo tanto, se debe colocar un modo de acceso que indica qué aplicaciones tienen acceso a la recopilación de preferencias, así como de operaciones que tienen permitido efectuar sobre ellas. Son tres modos que se puede elegir:

MODE_PRIVATE: Indica específicamente que tipo de aplicación presenta acceso a este tipo de preferencias.

MODE_WORLD_READABLE: Cada una de estas aplicaciones pueden leer estas preferencias, pero sólo la nuestra puede cambiarlas.

MODE_WORLD_WRITABLE: Cada una de las aplicaciones pueden leer y modificar estas preferencias.

2.3.12.5. Google Maps Android API

Se trata de una librería originada por Google que permite el uso de la geo-referenciación o geo-localización al interior de la aplicación móvil, su estructura está conformada por atributos, métodos y elementos que sirven para la interacción con el usuario, estos elementos ayudan a cambiar los mapas que existen obtenidos de Google Maps a los requerimientos del usuario, se debe tener en cuenta que el dispositivo va a servir de huésped por ello es necesario contener GPS, puesto que la librería interactúa de manera directa con el

hardware del dispositivo para hacer uso del conjunto de características que esta librería ofrece.

2.3.12.6. JSON

JSON que en sus siglas significan “Objeto de notación JavaScript” o en inglés “JavaScript Object Notation” consiste en un formato sencillo para el intercambio de información. Esta aplicación es fácil de leer, así como de escribir también, así como de la interpretación de la máquina. Está se fundamenta en un subconjunto de JavaScript. El formato de JSON es conocido en lenguajes como Java, PHP, JavaScript, C++, C#, Objective-C así como en otros. Esto hace que “JSON sea un lenguaje de un buen potencial para el cambio de datos” (Rhone, 2013, pág. 90).

El formato JSON es utilizado en un innumerable grupo de aplicaciones móviles, video juegos, web y otros. Este formato ésta constituida por 2 estructuras que se soportan en cualquier lenguaje de programación:

- Una colección de pares nombre/valor.
- Una lista ordenada de valores.

2.3.13. Streaming

El streaming constituye que mediante una palabra que demuestra alusión al escuchar música o ver videos sin necesidad de descargarlos, sino que permite utilizar los fragmentos de una forma secuencial a través de la red (como lo es Internet). Por ejemplo, “También se realiza el streaming cuando se difunde radio en vivo por Internet” (Castro, 2018).

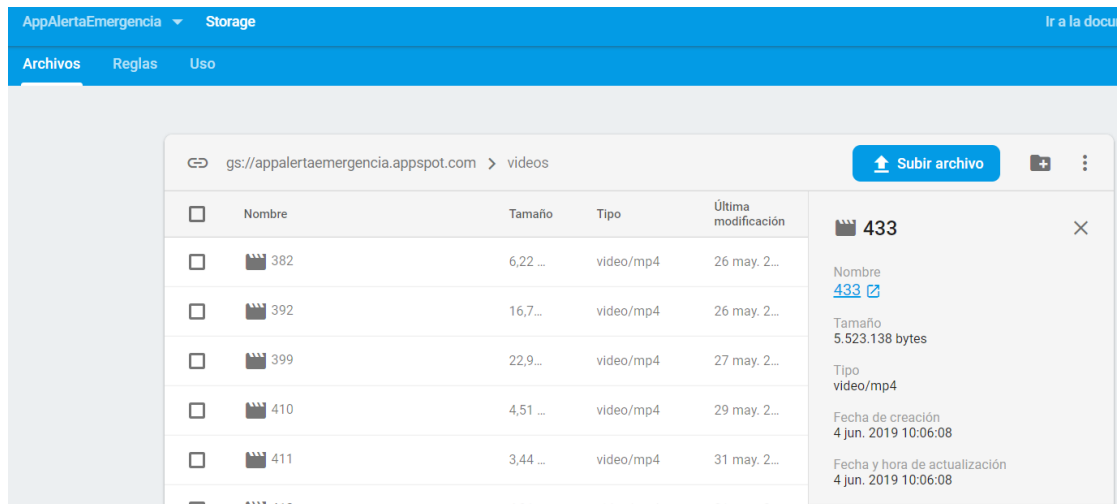


Figura 9. Muestra de videos en Firebase.

2.3.14. Dominio Web

Para la publicación en la web se usa los servidores que ofrecen el almacenamiento gratuito y a su vez está el dominio para ingresar a la web, el portal web de la tienda virtual de los perfiles cliente y administración están alojado de manera gratuita en el hosting 000webhost.com, que además de ser uno de los más inmensos de la red, ofrece a los usuarios 1.5 GB de almacenamiento y 100 GB de tráfico al mes para almacenar páginas web.

Además, cuenta una modalidad de pago en la cual el usuario puede ofrecer de más privilegios y numerosas características adicionales. 000webhost.com dispone de un panel de control que le permite administrar el sitio web y cuentas ftp que le ayuda hacer transferencias de archivos.



Figura 10. Página de inicio de Dominio Web.

Tomada de (000webhost, 2019).

2.3.15. **Firestore**

La aplicación Firestore se trata de una tecnología que ayuda a las aplicaciones web sin requerir de programación. Con la aplicación Firestore, no existe la necesidad de insistir en el sobre-provisionamiento de servidores o con la construcción de las API REST. Es compatible con la web, iOS, OS X, así como Android. Las aplicaciones que usan Firestore contribuyen a usar y controlar los datos, sin pensar en el almacenamiento de datos, y se sincronizan mediante algunos ejemplos de la aplicación en tiempo real. No existe la necesidad de escribir un código en el servidor (QUORA, 2016).

En cuanto a lo esencial de usar una Firestore, es que, para acceder al servicio web, no se requiere acceder a un servidor, puesto que la API sincroniza y guarda datos en la nube en tiempo real.

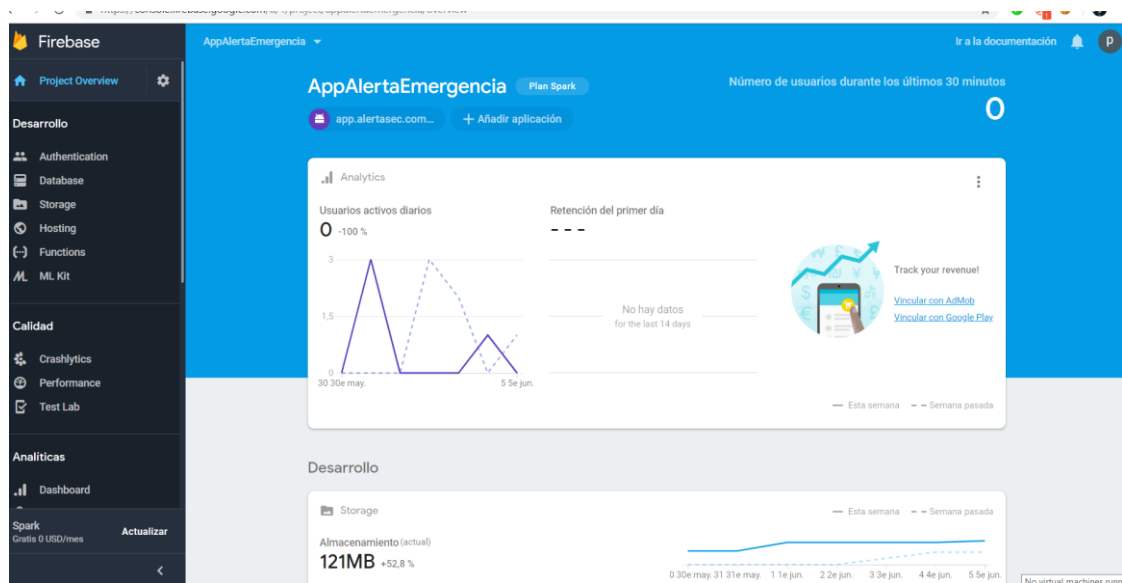


Figura 11. Página de inicio de Firebase

3. Capítulo III. Desarrollo del proyecto

3.1. Planteamiento del sistema de pánico

El sistema de pánico constituye el “Botón de Alerta” el cual es una nueva herramienta que permite alertar mediante un botón físico de los peligros que pueden ser objeto las personas en general (tercera edad, personas con capacidades especiales, etc.), ante el incremento de la delincuencia nacional e internacional que día a día es más notoria, tomando en cuenta que las Instituciones del Estado no han podido detener el auge delictivo, las cuales son llamadas a brindar protección y seguridad a la población, este sistema facilitará a miles de personas que se encuentren en estado de vulnerabilidad la alternativa para prevenir un delito así como constituirá una nueva herramienta en la lucha contra la inseguridad de la comunidad, cabe mencionar que ésta alerta se envía al encargado de monitoreo del servidor web. Una vez recibida la información, el operador actúa rápidamente comunicando a las entidades encargadas de prestar auxilio. En la siguiente figura se indica como un usuario envía una alerta al encargado de seguridad a través del sistema de alerta móvil, la misma que se puede observar en tiempo real y su ubicación.

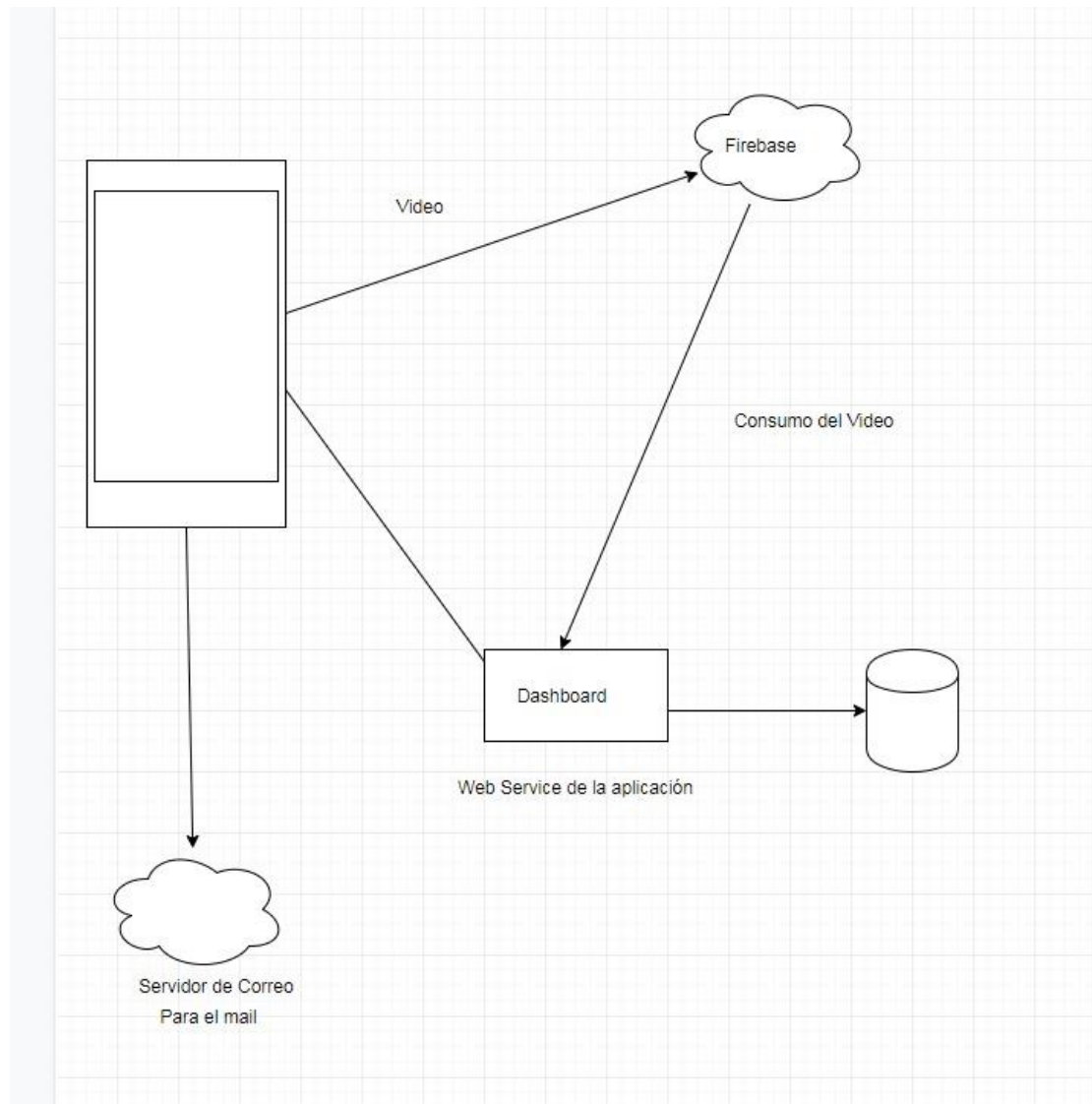


Figura 12. Planteo de la solución del sistema de pánico.

Este tipo de aplicación puede ser capaz de ejecutarse en un segundo plano, todo ello mediante una confirmación persistente que indica al usuario conocer que este servicio de acceso rápido se encuentra activo y que al momento de presionar dos veces de manera consecutiva el botón de volumen (+/-) manda una confirmación hacia el encargado de seguridad de manera que efectúe de forma eficaz una respuesta inmediata cuando una persona tenga una emergencia.

En lo que corresponde a los procesos de aplicación se especifican las entradas siguientes: el evento que se genera al momento cuando se produce una emergencia; así como también cuando el sistema de vigilancia que se

encuentra constantemente haciendo streaming del video. La información se la procesa en el interior de los servidores correspondientes y como resultado final se obtendrá el reporte, ya sea el encargado de seguridad de recibir la notificación en el centro de emergencias.

3.2. Diseño de la aplicación

3.2.1. Casos de uso

3.2.1.1. Diagrama de Casos de Uso General

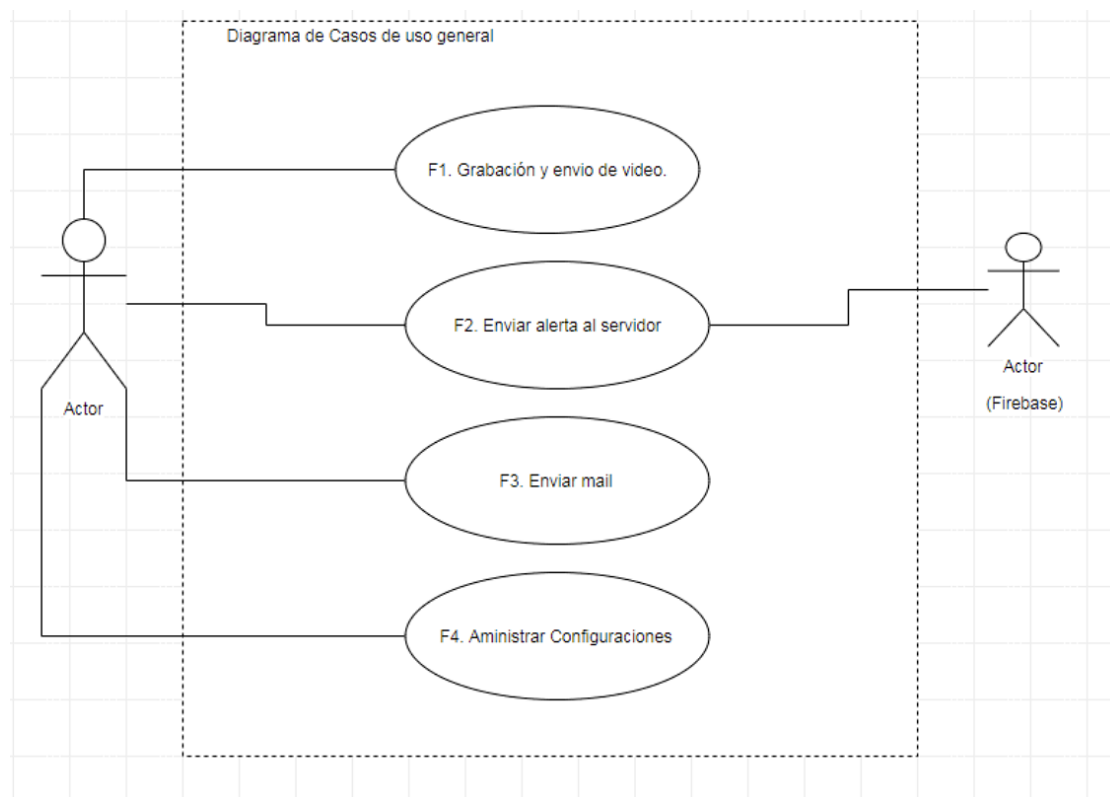


Figura 13. Diseño Casos de Uso General.

3.2.1.2. Diagrama de Casos de Uso Específico

3.2.1.2.1. F1. Grabación de video

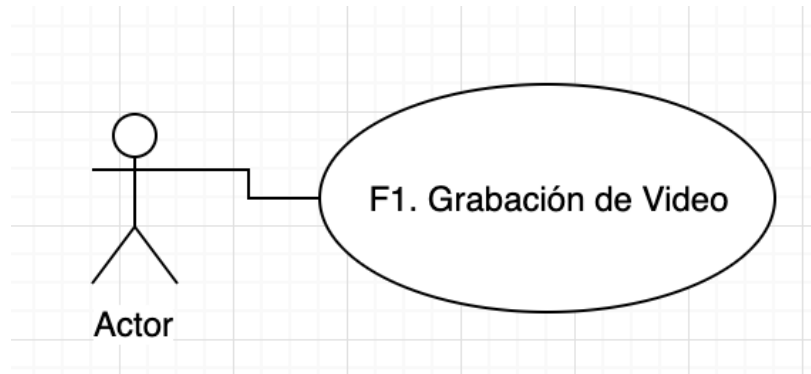


Figura 14. Diagrama Caso de Uso F1.

Tabla 3.

Funcionalidad F1.

Sistema:	Aplicación Alerta de Emergencias
Caso de uso:	Grabación de Video
Actor:	Usuario
Excepciones:	
<ol style="list-style-type: none"> 1. El actor selecciona "Acciona el botón de Generar alerta". 2. El sistema abre la activity de grabar. 3. El sistema el sistema graba durante el tiempo en el parámetro del sistema. 4. El sistema almacena el archivo en la memoria del dispositivo. (E1) 	
Flujo Alternativo:	
<ol style="list-style-type: none"> 5. Consultar con el administrador del sistema. 	
Excepciones:	Soluciones:
E1: El aplicativo no tiene los permisos para almacenar archivos.	Proporcionar los permisos en la configuración del teléfono.

3.2.1.2.2. F2. Enviar alerta al Servidor

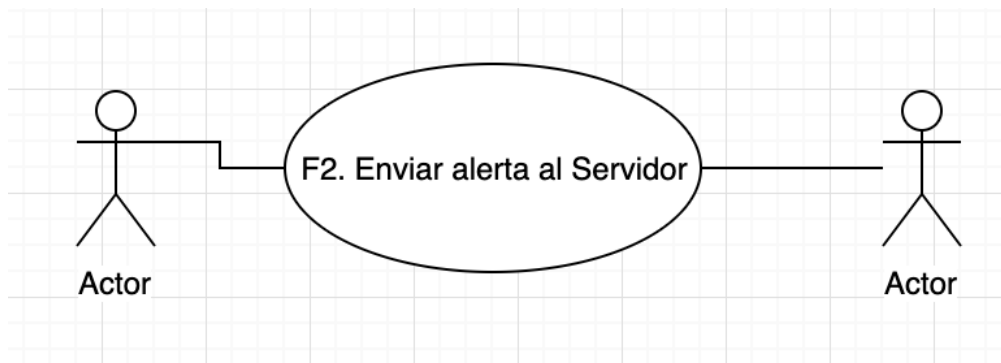


Figura 15. Diagrama Caso de Uso F2.

Tabla 4.

Funcionalidad F2.

Sistema:	Aplicación Alerta de Emergencias
Caso de uso:	Enviar alerta al Servidor
Actor:	Usuario
Excepciones:	
<ol style="list-style-type: none"> 1. El actor presiona el botón de subir al volumen. 2. El sistema abrirá el intent de video. 3. El sistema grabará automáticamente el video. 4. El sistema cargará los datos en el JSON que se enviará al servidor de alerta. 5. El sistema enviará el archivo al dashboard. (E1) 	
Flujo Alternativo:	
<ol style="list-style-type: none"> 6. Consultar con el administrador del sistema. 	
E1: Servicio Web no se encuentra disponible.	Contactar con el Administrador del sistema.

3.2.1.2.3. F3. Enviar Mail

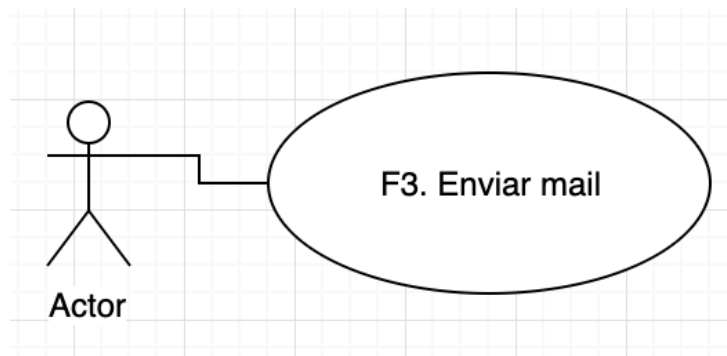


Figura 16. Diagrama Caso de Uso F3.

Tabla 5.

Funcionalidad F3.

Sistema:	Aplicación Alerta de Emergencias
Caso de uso:	Enviar mail
Actor:	Usuario
Excepciones:	
<ol style="list-style-type: none"> 1. El actor presiona el botón de subir el volumen o presionar el botón de enviar alerta. 2. El sistema accederá al mail que enviará las notificaciones. 3. El sistema generará el mail. 4. El sistema enviará el mail. (E1) 	
Flujo Alternativo:	
<ol style="list-style-type: none"> 5. Consultar con el administrador del sistema. 	
E1: Conexión a internet no disponible.	Contactar con el Administrador del sistema.

3.2.1.2.4. F4. Administrar configuraciones

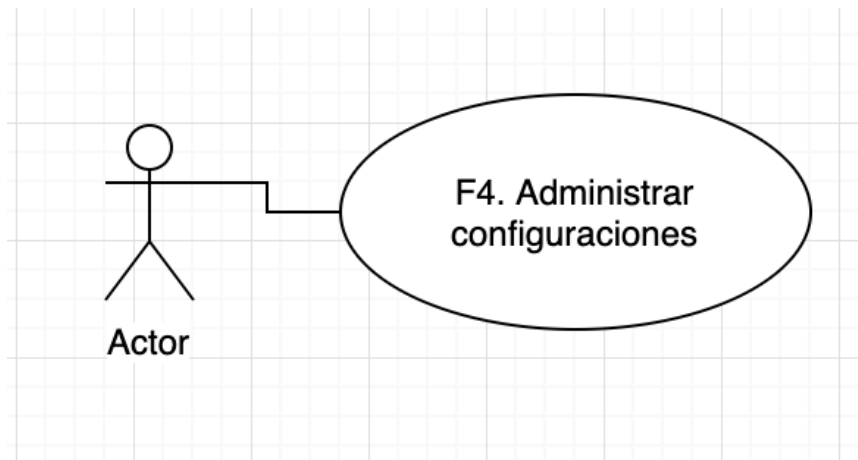


Figura 17. Diagrama Caso de Uso F4.

Tabla 6.

Funcionalidad F4.

Sistema:	Aplicación Alerta de Emergencias
Caso de uso:	Administrar configuraciones
Actor:	Usuario
Excepciones:	
<ol style="list-style-type: none"> 1. El actor presiona el botón de configuraciones. 2. El sistema abre el activity de configuraciones. 3. El actor ingresa las configuraciones y las guarda. 4. El sistema almacena las configuraciones. (E1) 	
Flujo Alternativo:	
<ol style="list-style-type: none"> 5. Consultar con el administrador del sistema. 	
E1: Conexión a la base no disponible.	Contactar con el Administrador del sistema.

3.2.1.3. Requerimientos Funcionales y No Funcionales

El objetivo de esta sección es proporcionar una descripción detallada de cada funcionalidad del aplicativo móvil.

3.2.1.3.1. Requerimientos Funcionales

Tabla 7.

Requerimientos funcionales.

Nº	Requerimiento
R1	El sistema será capaz de grabar de forma automática o bajo demanda, un archivo de video durante un tiempo específico.
R2	El sistema enviará un correo electrónico cuando el usuario genere una alerta tanto manual como automáticamente.
R3	El sistema permitirá configurar el tiempo de grabación como parámetro del sistema.
R4	El sistema permitirá enviar automáticamente el video almacenado en el teléfono.
R5	El sistema configura los datos de la cuenta que enviará y recibirá los correos electrónicos con la alerta.
R6	El sistema mostrará los datos de la alerta en un portal web.

3.2.1.3.2. Requerimientos No Funcionales

Tabla 8.

Requerimientos no funcionales.

Nº	Requerimiento
R7	El sistema debe correr en dispositivos con Sistema Operativo Android.

R8	El sistema debe correr en un servicio de segundo plano para receptor las peticiones del usuario.
----	--

3.3. Desarrollo de la aplicación

3.3.1. Especificaciones

En lo referente a las especificaciones se puede indicar que la interfaz de la aplicación del servicio de emergencia presenta un botón, el cual determina si se requiere o no mantener la aplicación corriendo, además se puede indicar que en un segundo plano se activará el evento de acceso rápido. Además de estos elementos se contará también con un dispositivo que envía directamente la alerta, siendo no necesario el presionar el botón de pánico.

Cuando el botón cambia de color (de off a on) se aplica una notificación en el interior del canal de notificaciones de Android, el cual permite a la aplicación sostenerse en segundo plano, a la escucha de que la acción sea terminada y luego de lo cual se envía la notificación para los destinatarios.

Se especificará un contador entorno al cual se validará si los botones de volumen (+/-) fueron accionados en un intervalo de tiempo, y al finalizar este tiempo el contador volverá a cero y esperará a que la acción sea ejecutada para iniciar otra vez con la cuenta.

Se desarrolló para esta aplicación un método dentro del activity ButtonService, el mismo que es el encargado, cuando se realizan las acciones de VOLUME_UP y VOLUME_DOWN aplicando un ciclo de pulsación del botón de volumen, el cual simulará una acción fundamental y con ello se realice el envío del mensaje.

Cuando el receiver ha logrado obtener la información la cual es capturada en el evento, se realizará la entrega de la información hacia el servidor para alertas por medio de smartphone el mismo que está en las instalaciones del encargado de seguridad.

Si bien toda la información, por parte del usuario como de indicaciones médicas como contacto de emergencia se entregará en formato JSON luego se lo enviará hacia el destino.

Se puede especificar además que este tipo de aplicación mantendrá con un hilo de envío que contribuirá a la comunicación que se da entre cliente y servidor. Es mediante lo cual se permite a la aplicación realizar una petición de lectura de datos hacia el servidor base.

En referencia al funcionamiento del proceso en un sentido amplio será explicado de forma detallada en la figura 18.

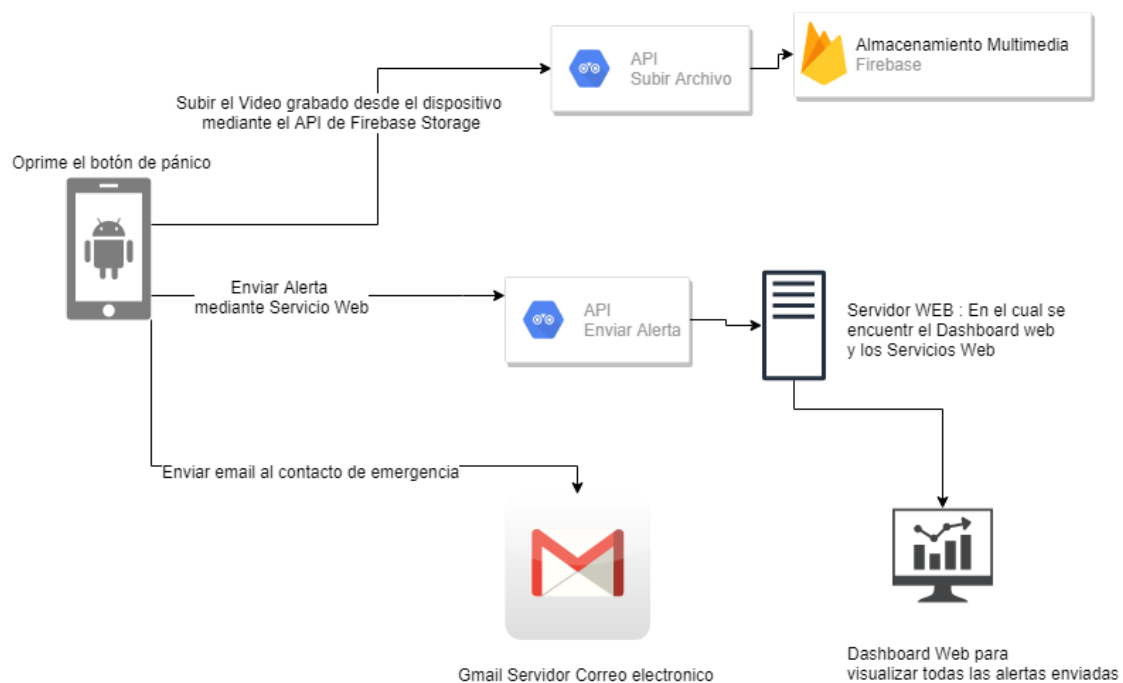


Figura 18. Diagrama explicativo sobre la aplicación.

3.4. Implementación de la aplicación

3.4.1. Backend

En este apartado se podrá apreciar la configuración de los servidores que se usaron para el almacenamiento de datos y multimedia en la Nube (BDD, Firebase y Servidor Web).

3.4.1.1. Firebase

Para poder almacenar los videos que se graban en una alerta mediante el dispositivo se utilizó el servicio Firebase Storage.

Para lo cual se tiene que crear un proyecto en Firebase, en este caso se llamó AppAlertaEmergencia, el mismo que debe ser enlazado a la aplicación móvil.



Figura 19. Interfaz Firebase.

Para poder acceder a la ubicación dentro de la cual se guardan los videos nos tenemos que dirigir a la opción /Storage y dirigirnos a la carpeta videos.

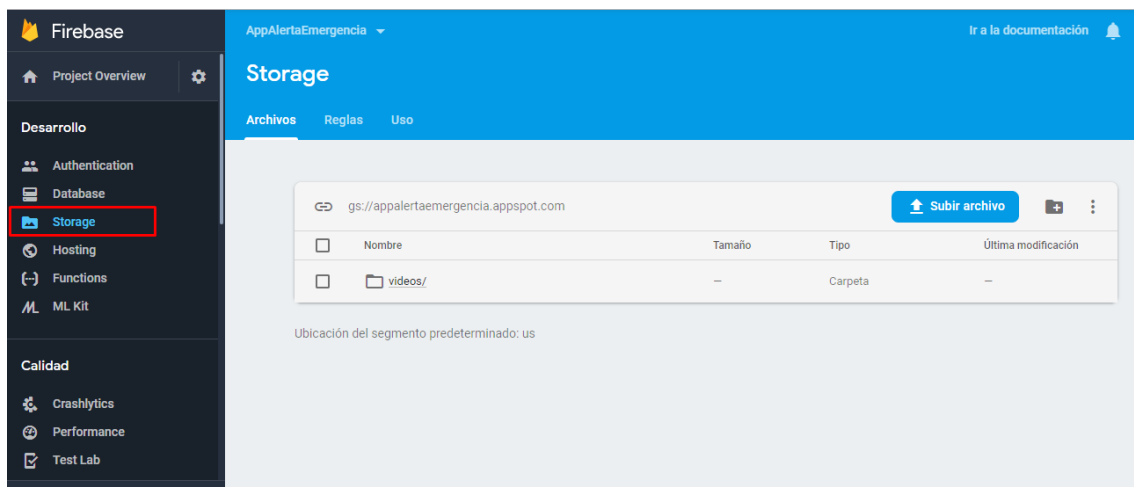


Figura 20. Interfaz Firebase /Storage.

Ahora después de abrir la carpeta /videos podemos observar todos los videos que se han grabado desde el dispositivo, los cuales se guardan con el (ID) de la alerta que se encuentra en la base de datos.

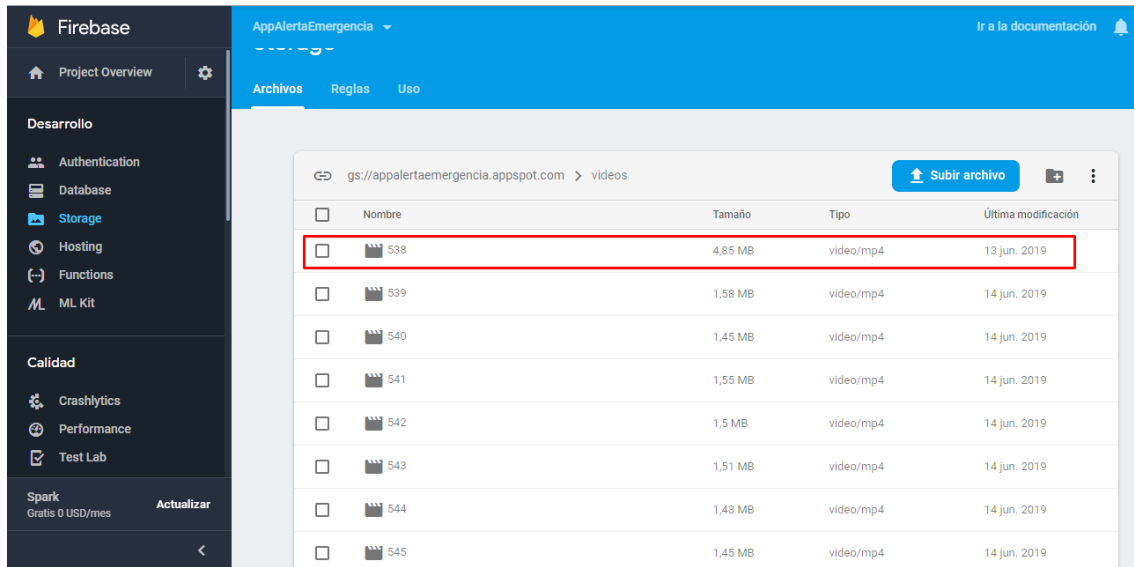


Figura 21. Interfaz Firebase (ID del usuario).

3.4.1.2. Servidor Web

Para el servidor Web se usó <https://www.000webhost.com/> que permite poseer servidores web de forma gratuita así como bases de datos, las tecnologías que se usaron en el servidor fueron:

- PHP como lenguaje de backend
- MYSQL como motor de base de datos
- HTML y Javascript para la implementación del dashboard web

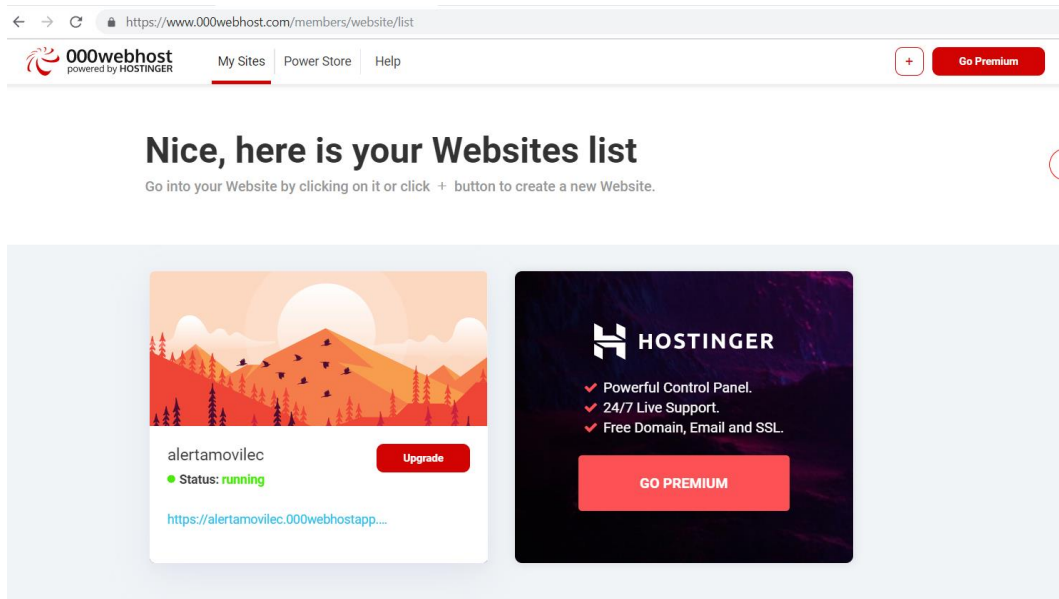


Figura 22. Interfaz HTML (dashboard web).

- Base de datos

Para acceder a la base de datos se debe dirigir a la opción de Tools/Database Manager, después de lo cual si escogemos la base de datos resaltada podremos acceder a phpmyadmin que nos permitirá manejar la base de datos.

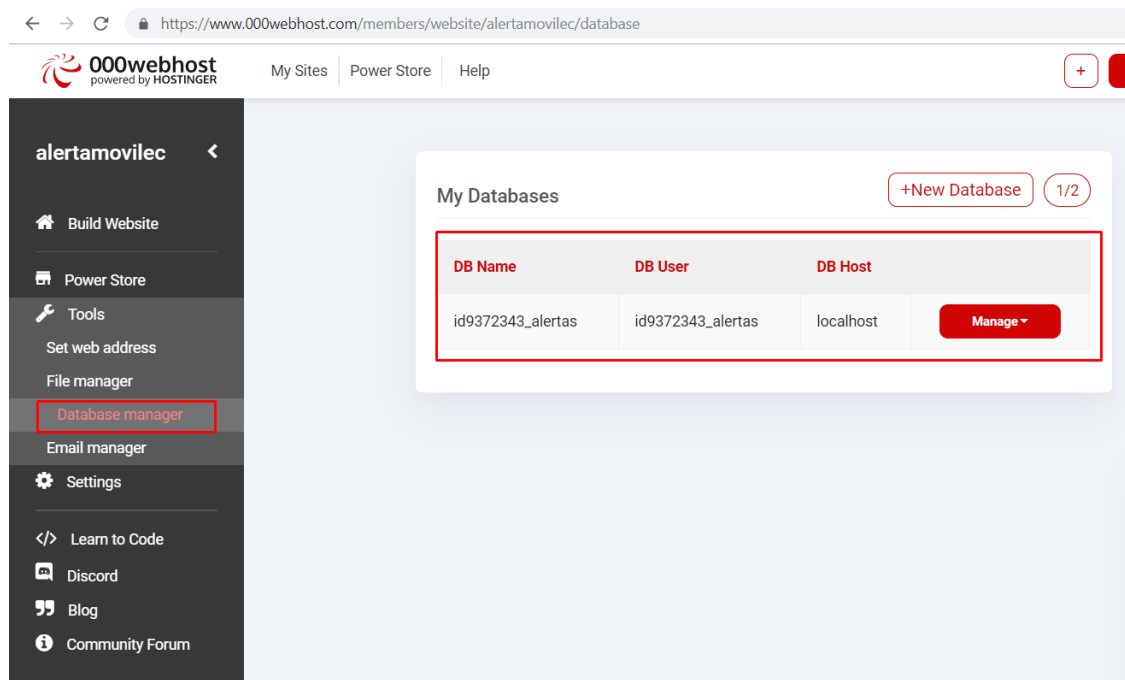


Figura 23. Interfaz Base de Datos.

The screenshot shows the phpMyAdmin interface for a database named 'id9372343_alertas'. The table 'alertas' is selected, and its data is displayed in a table format. The table has columns for 'codigo', 'usuario', 'contacto', 'fecha', 'latitud', and 'longitud'. The data includes 18 rows, each with a unique 'codigo' and associated user and contact information, along with a timestamp and geographic coordinates.

codigo	usuario	contacto	fecha	latitud	longitud
1	juan	pedro gonzales	2019-04-21 06:00:00	0.15	-78.233
2	juan	pedro gonzales	2019-04-21 07:00:00	0.045	-78.133
3	usertestapi	contacto test	2019-04-28 16:53:30	-0.12922	-78.499687
4	test	contacto	2019-04-29 00:10:01	0	-78.987
5	test	contacto	2019-04-29 00:10:09	0	-78.987
7	edison	edison López/09998	2019-04-29 00:16:38	0	0
8	edison	edison López/09998	2019-04-29 00:23:10	0	0
9	edison	edison López/09998	2019-04-29 04:44:40	0	0
10	edison	edison López/09998	2019-04-29 14:30:48	-0.2039464	-78.4830326
11	edison	edison López/09998	2019-04-29 14:31:01	-0.2039464	-78.4830326
12	Erick	Ermelinda/0967790420	2019-04-29 20:12:29	-0.0778469	-78.4683472
13	Erick	Ermelinda/0967790420	2019-04-29 20:17:21	0	0
14	Erick	Ermelinda/0967790420	2019-04-29 21:17:39	-0.0778469	-78.4683472
15	Erick	Ermelinda/0967790420	2019-04-30 01:11:40	-0.0778469	-78.4683472
16	marios	erick c/0999203689	2019-04-30 12:57:41	-0.1702893	-78.4721497
17	marios	erick c/0999203689	2019-04-30 12:59:03	-0.1699906	-78.4718902
18	dario	joel lopez/0990313166	2019-04-30 13:00:21	-0.1699906	-78.4718902

Figura 24. Interfaz PHP.

- Dashboard Web y API REST (Servicio Web)

The screenshot shows the 000webhost dashboard for a website named 'alertamovilec'. The interface is clean and modern, with a dark sidebar on the left and a light main content area. The 'File manager' option is highlighted in the sidebar. The main content area features a large white box with a cloud upload icon and the text 'Fast, Easy & Secure File Upload'. Below this, there is a red button labeled 'Upload files now' and a smaller red button labeled 'Let's do it'.

Figura 25. Interfaz Dashboard Web.



The screenshot shows a web hosting control panel interface for '000webhost'. The interface has a red header bar with the logo and navigation icons. Below the header, there is a sidebar on the left with a tree view showing folders like 'public_html' and 'tmp'. The main area displays a table of files with columns for Name, Size, Date, and Permissions.

Name	Size	Date	Permissions
.htaccess	0.2 kB	2019-04-22 03:22:00	-rw-r--
api_alerta.php	2.2 kB	2019-05-20 05:52:00	-rw-r--
dashboard.php	3.0 kB	2019-04-28 17:44:00	-rw-r--

Figura 26. Interfaz API REST.

3.4.2. Frontend

A continuación, podremos ver las interfaces Android que se desarrollaron y las cuales se comunican con el BACKEND mediante Servicios Web.

3.4.2.1. Interfaz de inicio

Desde la cual se puede acceder a la configuración de la aplicación, obtener la ubicación del usuario y también enviar la alerta la misma que se enviará al servidor, a una cuenta de correo y además subirá a Firebase Storage un video-grabado desde el dispositivo.

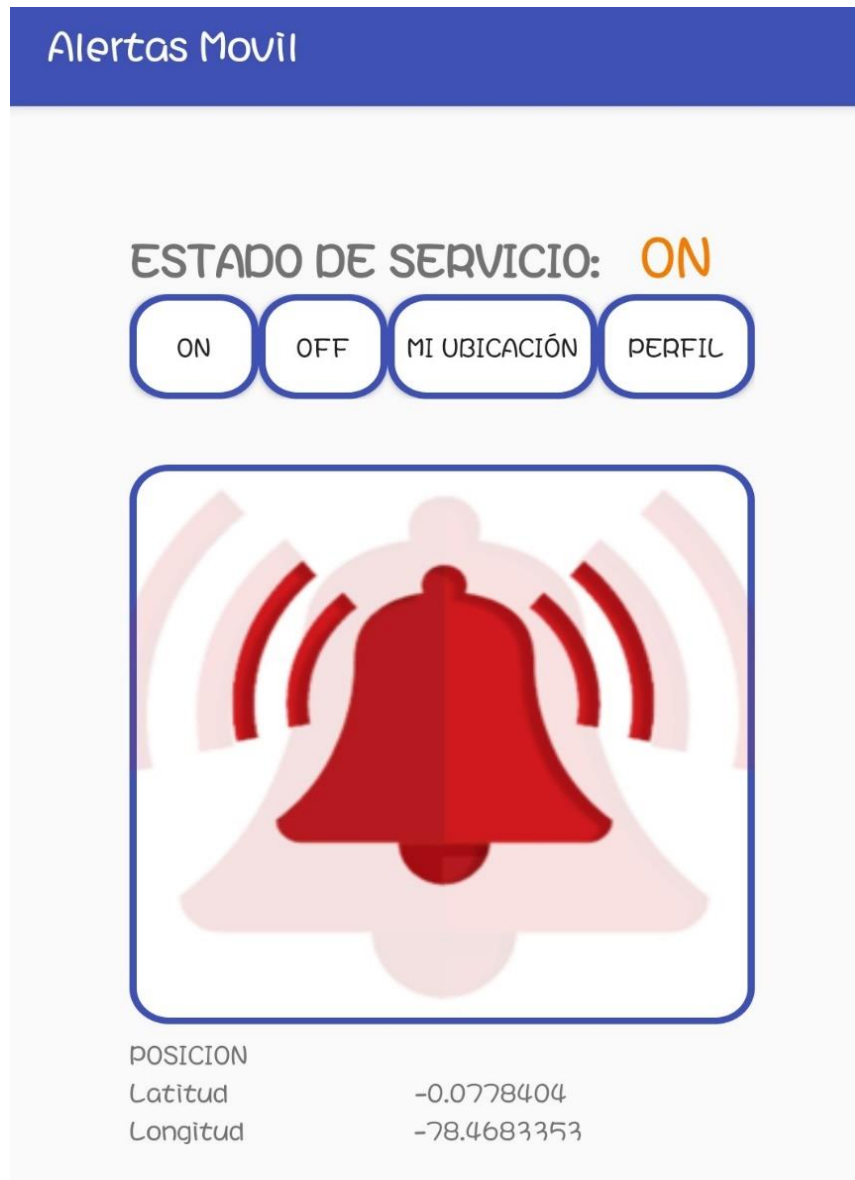


Figura 27. Interfaz principal de la APP.

3.4.2.2. Interfaz de comunicación

En esta pantalla se configurará el nombre del usuario de la aplicación, los datos para enviar el correo al contacto de emergencia y además la duración en milisegundos del video que se grabará automáticamente una vez enviada la alerta.

← Configuración Contacto

Usuario

Tu nombre

Tu cédula

Tu número celular

Tipo Sangre(O+)

O+

Discapacidad

Visual

Contacto Emergencia

Nombre completo contacto

Correo electrónico contacto

Teléfono contacto

Duración de video (Segundos)

3s

GUARDAR

Figura 28. Interfaz Registro de datos.

3.4.2.3. Interfaz de Grabar Video

Misma que se abrirá automáticamente una vez que el usuario oprima el botón de enviar alerta la cual después de terminar de grabar el video se comunicará con Firebase para poder subir el video.

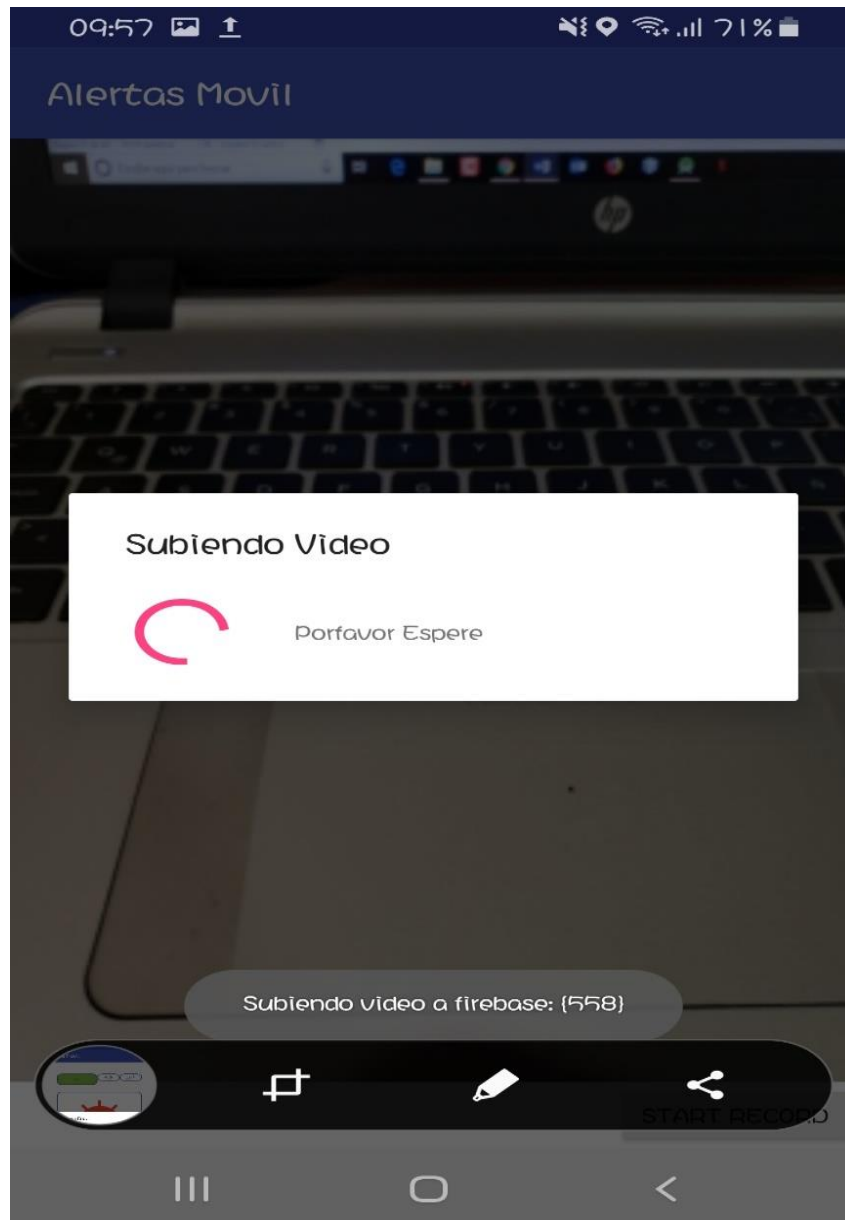


Figura 29. Interfaz Envío Video al Firebase.

3.5. Funcionamiento de la aplicación

De acuerdo a lo que se puede determinar en la figura 12, cuando se procede instalar la aplicación se solicitará que los datos sean ingresados por parte del usuario: Nombre y Apellido, además se solicita información adecuada del contacto de emergencia, nombre completo, correo electrónico y número telefónico, todo ello se realiza para estar comunicados con la persona cuando

se pueda producir algún percance con el usuario, que estos datos sean ingresados y enviados hacia el encargado de seguridad.

Configuración Contacto

Usuario

Tu nombre

Tu Cedula

Tipo Sangre(O+)
O+

Discapacidad
Auditiva

Contacto Emergencia

Nombre Completo Contacto

Correo electrónico Contacto

Telefono Contacto

Duración de Video (Milisegundos)
Duración de Video Milisegundos (5000 Default)

GUARDAR

Figura 30. Registro de datos “usuario y contacto emergencia”.

La figura 31 indica el diseño que demuestra la aplicación, en la cual el activity principal mostrará las funcionalidades, tenemos el botón (on/off) el cual activa el servicio y un botón con la imagen de alerta que se usará para entregar la notificación si se quiere.

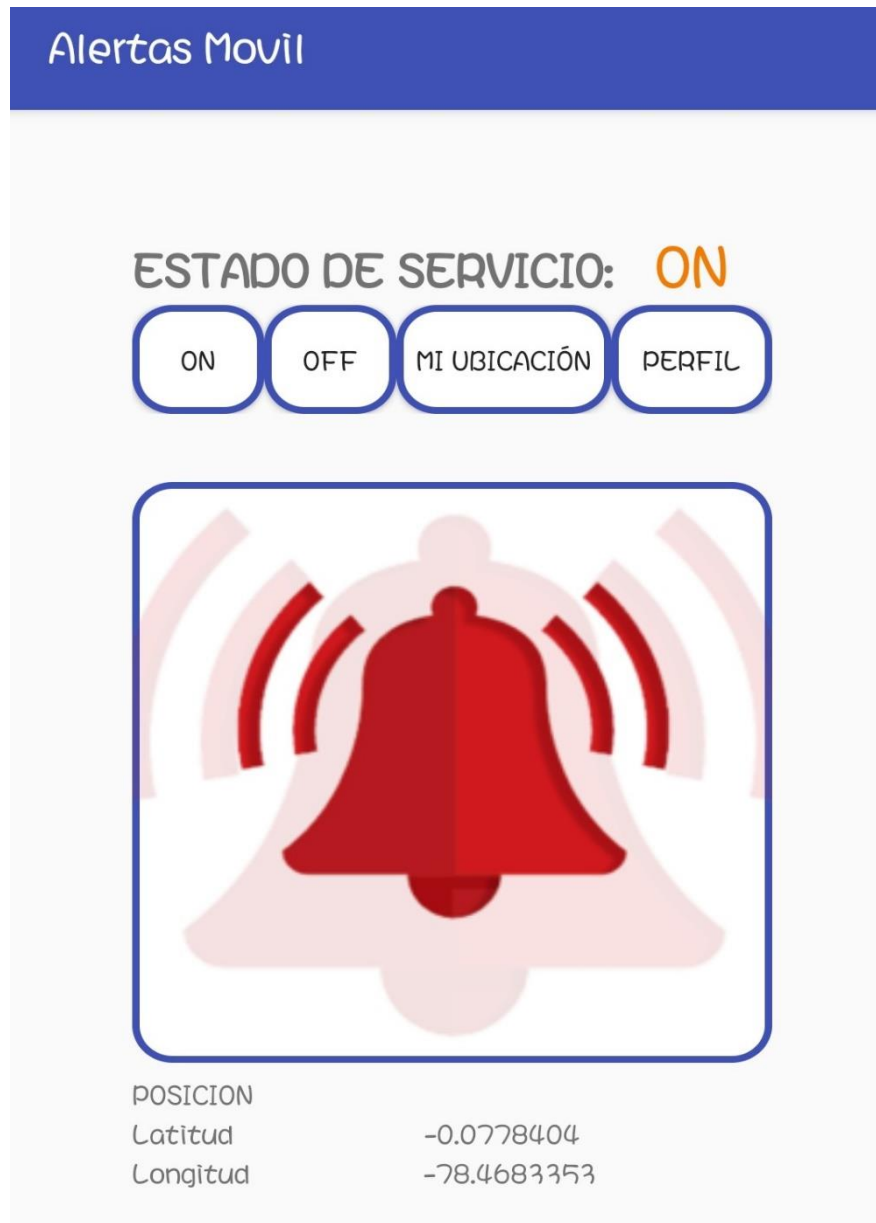


Figura 31. Presentación principal de la aplicación.

De acuerdo a la figura 14, el usuario puede optar por enviar la emergencia aplastando el botón imagen de envío o luego el botón (on/off) iniciará la opción de acceso rápido con lo cual se confirmará los servicios de emergencia manteniendo aplastada una vez los botones de volumen.

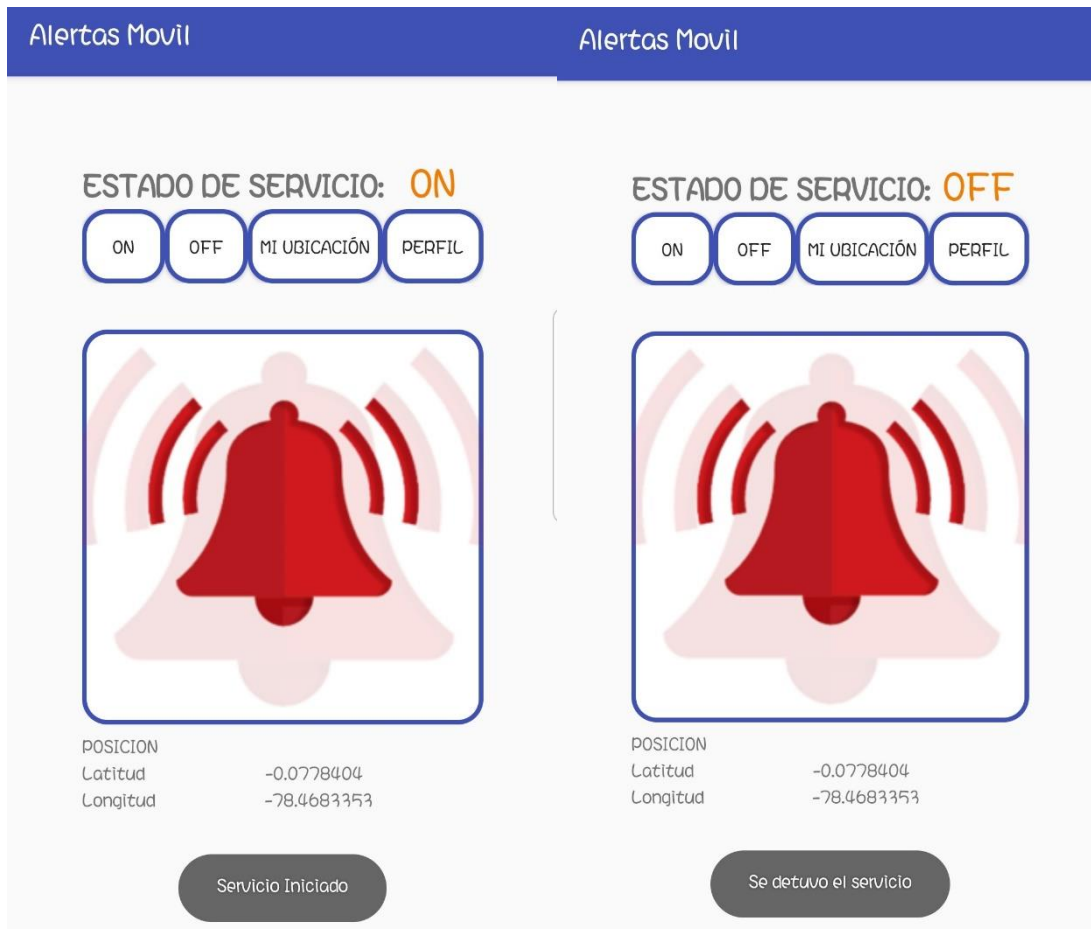


Figura 32. Servicio activo y servicio inactivo.

En el momento en que el servicio es utilizado, se puede determinar que la aplicación será la que se encargue del envío de las notificaciones entorno a los servidores siguientes:

- Servidor Web (Dominio Web)
- Servidor Firebase

Luego de ser realizado esto se puede desplegar la notificación la cual se indica en la figura 15 y con ello se informa que la emergencia ha sido confirmada de manera óptima hacia el encargado de seguridad y hacia el servidor web.

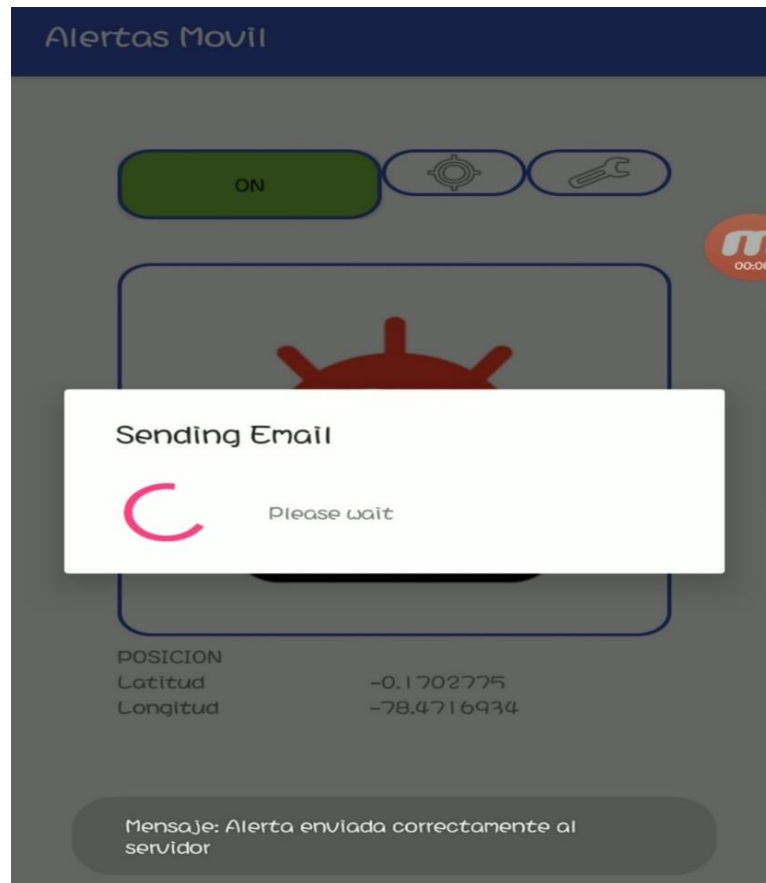


Figura 33. Presentación de resultado afirmativo de envío de correo electrónico.

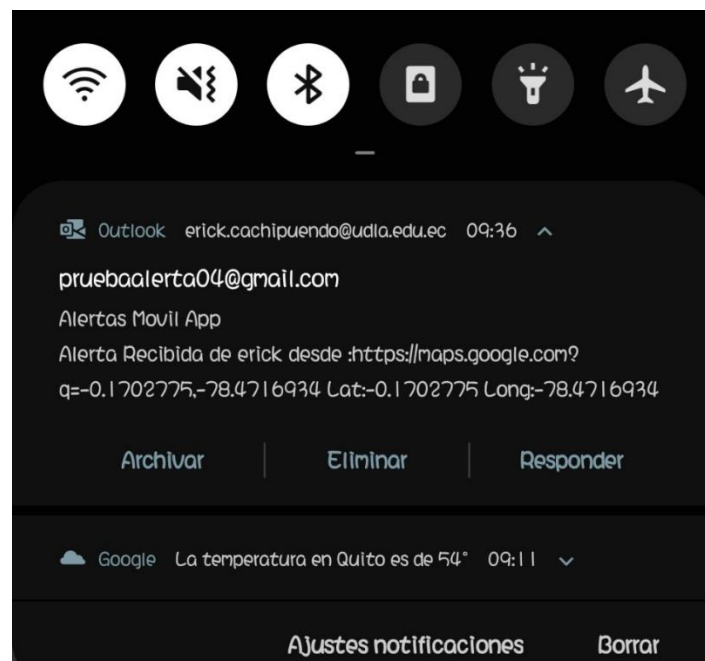


Figura 34. Información de correo electrónico enviado al encargado de seguridad y a la tercera persona de contacto de emergencia con información de la persona afectada.

En la figura 34 se puede indicar que el correo presenta la información de la persona y la ubicación del evento.

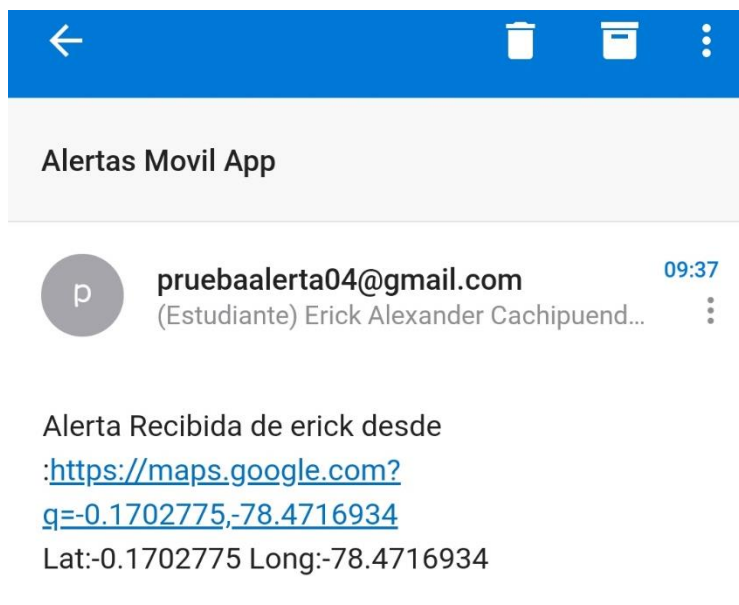


Figura 35. Información del correo electrónico mandado al encargado presentando el enlace de la ubicación.

En la figura 36 se puede indicar como llegan las alertas en el servicio web.

428	edison	joss/0998883047	2019-05-31 14:09:41	Lugar
429	erick	erme lopez/0990313166	2019-05-31 15:57:24	Lugar
430	erick	erme lopez/0990313166	2019-05-31 15:58:28	Lugar
431	erick	jorge y/0988552244	2019-05-31 16:20:23	Lugar
432	erick	jorge y/0988552244	2019-05-31 16:21:10	Lugar
433	erick	jorge y/0988552244	2019-06-04 14:05:41	Lugar
434	erick	jorge y/0988552244	2019-06-07 03:45:28	Lugar
435	edison	joss/0998883047	2019-06-07 11:12:57	Lugar
436	erick	jorge y/0988552244	2019-06-07 13:33:54	Lugar
437	erick	jorge y/0988552244	2019-06-07 13:37:36	Lugar

Powered by  000webhost

Figura 36. Dashboard web.

4. Capítulo IV. Análisis de resultados y pruebas

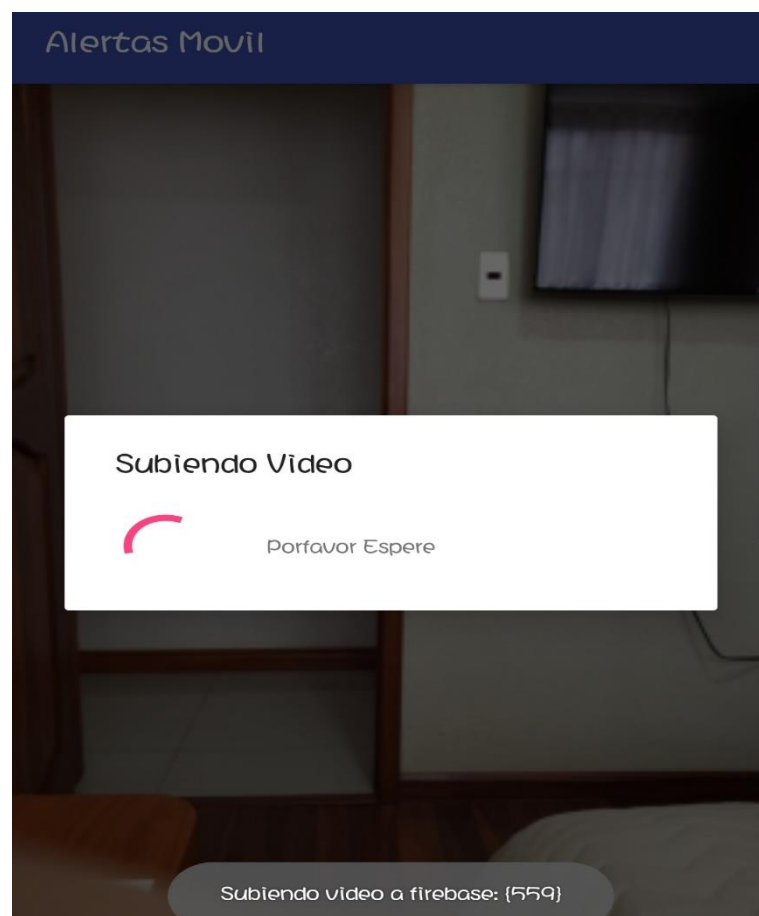
A continuación, se presenta los resultados en función de los requerimientos del sistema:

4.1. Requerimientos funcionales

Tabla 9.

Resultados en función de los requerimientos funcionales.

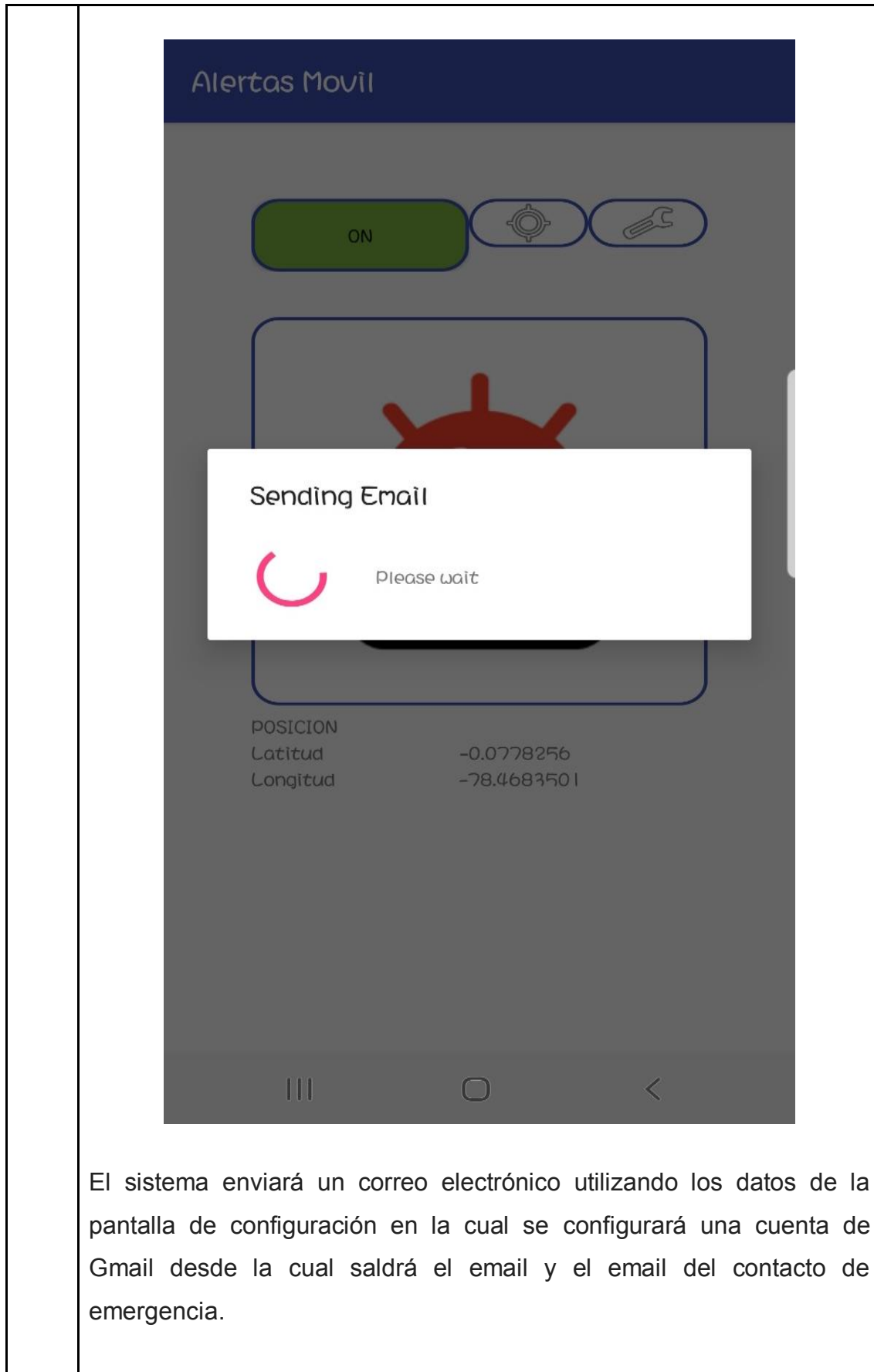
Nº	Requerimiento
R1	El sistema será capaz de grabar de forma automática o bajo demanda un archivo de video durante un tiempo específico. RESULTADO:



El video se graba automáticamente por un tiempo determinado que puede ser configurado en las configuraciones de la app, después de que se termine de grabar el video se guarda en la carpeta raíz del dispositivo y automáticamente se subirá a Firebase a una carpeta llamada videos como se puede ver a continuación.

File Name	Size	Format	Date
549	1,54 MB	video/mp4	14 jun. 2019
550	1,66 MB	video/mp4	14 jun. 2019
552	1,44 MB	video/mp4	14 jun. 2019
553	1,51 MB	video/mp4	14 jun. 2019
554	1,43 MB	video/mp4	14 jun. 2019
555	1,55 MB	video/mp4	14 jun. 2019
556	1,6 MB	video/mp4	14 jun. 2019
557	2,5 MB	video/mp4	18 jun. 2019
558	1,43 MB	video/mp4	19 jun. 2019
559	318,7 KB	video/mp4	19 jun. 2019
DEFAULT_NAME	8,5 MB	video/mp4	21 may. 2019

R2	<p>El sistema enviará un correo electrónico cuando el usuario genere una alerta tanto manual como automáticamente.</p> <p>RESULTADO:</p>



Configuración Contacto

Usuario

Tu nombre

Tu Cedula

Tipo Sangre(O+) **O+** ▼

Discapacidad **Auditiva** ▼

Contacto Emergencia

Nombre Completo Contacto

Quién recibe el email

Correo electronico Contacto

Telefono Contacto

Duracion de Video (Milisegundos)

GUARDAR

Una vez enviada la alerta se recibirá un correo con la ubicación del usuario que envió la alerta, que al momento de hacer click podremos ver mediante Google Maps el lugar de la alerta:

←
🗑️
📧
⋮

Alertas Movil App

P


pruebaalerta04@gmail.com 10:00

(Estudiante) Erick Alexander Cachipuend... ⋮

Alerta Recibida de Erick Prueba desde

[:https://maps.google.com?q=-0.0778256,-78.4683501](https://maps.google.com?q=-0.0778256,-78.4683501)

Lat:-0.0778256 Long:-78.4683501

R3	<p>El sistema permitirá configurar el tiempo de grabación como parámetro del sistema.</p> <p>RESULTADO:</p> <p>En la opción de configuración de la app se puede configurar el tiempo en milisegundos durante el cual se grabará el video automáticamente.</p> 
R4	<p>El sistema permitirá enviar automáticamente el video almacenado en el teléfono.</p> <p>RESULTADO:</p> <p>Después de grabar el video este se alojará en la nube de Firebase en una carpeta llamada /videos y con el ID de la alerta que consta en la base de datos, como se ve a continuación.</p>

The image shows the Firebase Storage console interface. The top navigation bar includes the Firebase logo, the project name 'AppAlertaEmergencia', and the 'Storage' section. A sidebar on the left lists various development tools like Authentication, Database, Storage, Hosting, Functions, and ML Kit. The main area displays a table of stored files.

File Name	Size	Type	Created
549	1,54 MB	video/mp4	14 jun. 2019
550	1,66 MB	video/mp4	14 jun. 2019
552	1,44 MB	video/mp4	14 jun. 2019
553	1,51 MB	video/mp4	14 jun. 2019
554	1,43 MB	video/mp4	14 jun. 2019
555	1,55 MB	video/mp4	14 jun. 2019
556	1,6 MB	video/mp4	14 jun. 2019
557	2,5 MB	video/mp4	18 jun. 2019
558	1,43 MB	video/mp4	19 jun. 2019
559	318,7 KB	video/mp4	19 jun. 2019
DEFAULT_NAME	8,5 MB	video/mp4	21 may. 2019

The file '559' is highlighted with a red border. A modal window provides details for this file:

- Nombre:** 559
- Tamaño:** 326.349 bytes
- Tipo:** video/mp4
- Fecha de creación:** 19 jun. 2019 10:00:40
- Fecha y hora de actualización:** 19 jun. 2019 10:00:40
- Ubicación del archivo:** (dropdown menu)
- Otros metadatos:** (dropdown menu)

At the bottom of the screenshot, a video player is visible, showing a close-up of a laptop keyboard. The video progress is at 0:03 / 0:03.

R5

El sistema configura los datos de la cuenta que enviará y recibirá los correos electrónicos con la alerta.

RESULTADO:

EL usuario mediante la pantalla de configuración puede parametrizar el nombre del usuario de la alerta, los datos del contacto de emergencia, la cuenta de Gmail desde la cual se envían los correos y además el tiempo de duración del video.

13:27

← Configuración Contacto

Usuario

Erick Cachipundo

1718546920

0967790420

Tipo Sangre(O+)

O+

Discapacidad

Auditiva

Contacto Emergencia

Ermelinda López

ecachipundo@gmail.com

0967790420

Duración de video (Segundos)

3s

GUARDAR

R6

El sistema mostrará los datos de la alerta en un portal web.

RESULTADO:

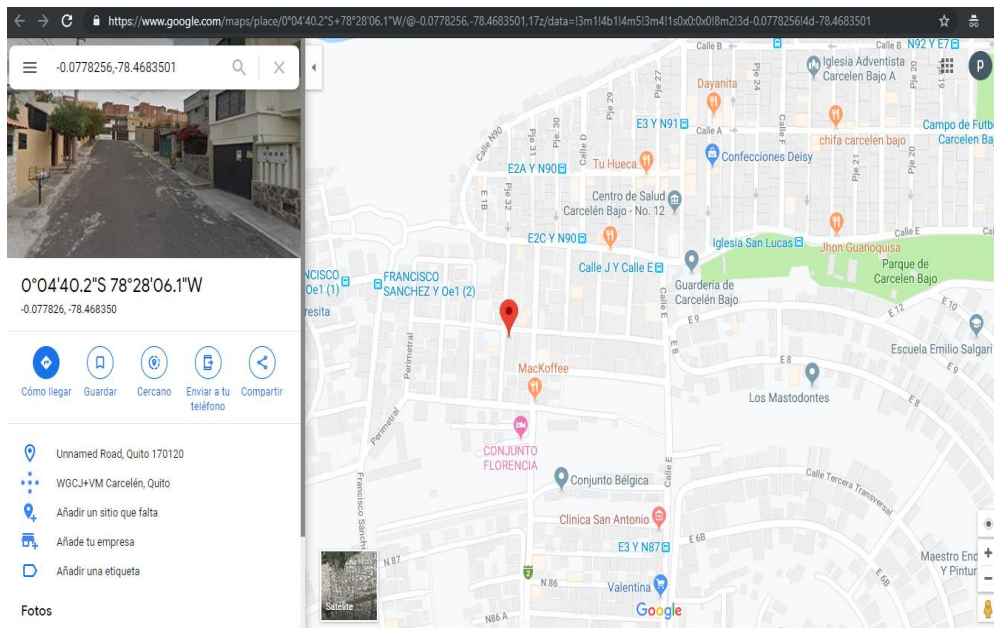
Se implementó un dashboard web alojado en la siguiente dirección <https://alertamovilec.000webhostapp.com/dashboard.php> el cual nos permite ver el registro de todas las alertas que se han enviado desde el dispositivo con la información del usuario y del contacto de emergencia, además se puede ver el lugar desde el cual se envió la alerta mediante Google Maps.

Alertas

Show 10 entries Search:

Codigo	Usuario Emergencia	Cedula Usuario	Tipo de Sangre	Discapacidad	Contacto Emergencia	Fecha	Ubicacion
622	Erick Cachipueno	1718546920	A+	Fisica	juan test/0967790420	2019-07-11 22:07:41	Lugar
621	Erick Cachipueno	1718546920	A+	Fisica	juan test/0967790420	2019-07-11 22:07:40	Lugar
620	Erick Cachipueno	1718546920	A+	Fisica	juan test/0967790420	2019-07-11 22:07:40	Lugar
619	Erick Cachipueno	1718546920	A+	Fisica	juan test/0967790420	2019-07-11 22:01:31	Lugar
618	Erick Cachipueno	1718546920	A+	Fisica	juan test/0967790420	2019-07-11 21:51:15	Lugar
617	Erick Cachipueno	1718546920	A+	Fisica	juan test/0967790420	2019-07-11 21:29:41	Lugar
616	usertestapi	1714678923	B	ninguna	contacto test	2019-07-11 20:37:16	Lugar
615	Erick Cachipueno				patricio/0999203666	2019-07-11 13:18:55	Lugar
614	Erick Cachipueno				patricio/0999203666	2019-07-10 22:30:02	Lugar
613	Erick Cachipueno				patricio/0999203666	2019-07-10 22:28:58	Lugar

Showing 71 to 80 of 691 entries Previous 1 ... 7 8 9 ... 70 Next



4.2. Requerimientos no funcionales

Tabla 10.

Resultados en función de los requerimientos no funcionales.

N°	Requerimiento
R7	<p>El sistema debe correr en dispositivos con Sistema Operativo Android.</p> <p>RESULTADO:</p> <p>La aplicación móvil fue desarrollada exclusivamente para dispositivos Android utilizando Android Studio el cual nos permite crear aplicaciones nativas para Android.</p>
R8	<p>El sistema debe correr en un servicio de segundo plano para receptor las peticiones del usuario.</p> <p>RESULTADO:</p> <p>La aplicación móvil aunque se cierre seguirá escuchando el evento del botón para subir el volumen para determinar si se tiene que enviar la alerta o no, esto se logró mediante la creación de un Servicio Android.</p>

5. CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

Se concluyó el desarrollo de una aplicación móvil para el proyecto ALERTA DE EMERGENCIAS, con el fin de proporcionar a los ciudadanos una herramienta para alertar y documentar posibles incidentes o emergencias.

Se desarrolló una aplicación para teléfonos inteligentes con sistema operativo Android, para lo cual se empleó Java como lenguaje de programación y el IDE Android Studio que es el recomendado por el fabricante para la construcción de aplicaciones nativas.

Estas herramientas prestaron a lo largo de todo el proceso un sin número de recursos y funcionalidades que sirvieron de soporte. La documentación existente es sumamente amplia y específica para diferentes casos de uso, apoyado de una comunidad creciente.

Los diagramas UML fueron de gran apoyo durante el proceso en las fases iniciales, a fin de tener un entendimiento común sobre la arquitectura del sistema y sus funcionalidades.

El servicio en segundo plano nos brindó una herramienta para poder monitorear la acción de botones físicos del teléfono para ejecutar acciones específicas.

Se consiguió el desarrollo de una aplicación móvil para teléfonos inteligentes que permiten enviar alertas hacia un servidor junto con un video al proveedor de almacenamiento "Firebase" que ayuda a observar los incidentes de una manera más detallada.

Se logró determinar los principales eventos que suceden en las calles de nuestro país como: asaltos, robos, accidentes, etc., este proyecto tiene la expectativa de facilitar el seguimiento para reducir los actos delictivos,

mejorando los tiempos de asistencia a accidentes que se puedan presentar con adultos mayores y la comunidad en general.

La ubicación precisa de la persona que tiene la emergencia o incidente emitida por el sistema, depende de la gama del teléfono que se esté utilizando (baja, media o alta).

Para suplir esta necesidad de inseguridad en la sociedad ecuatoriana nace la idea del diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente, basada en el sistema operativo Android.

Para el desarrollo de una aplicación móvil bajo el sistema operativo Android se utilizó el modelo en cascada, que consiste en: requisitos del software, diseño, implementación, verificación e instalación y mantenimiento. Se hace necesario realizar pruebas a nivel de diseño del uso de la aplicación a través de las técnicas de prototipado.

5.2. Recomendaciones

Es imprescindible que el mantenimiento de la aplicación se debe realizar en tres aspectos fundamentales; contenido, diseño y funcionalidad. El aplicativo en sus próximas versiones puede implementar nuevos métodos de seguridad o visualización de datos junto al análisis de las estadísticas que brinda el aplicativo.

Se recomienda en próximas versiones el realizar la aplicación para otras plataformas móviles ya que la seguridad debe ser brindada a todos por igual.

Es recomendable usar un dispositivo con sistema operativo mayor a Android (API 23), debido a que existe compatibilidad y mejoras con respecto a los métodos utilizados en el desarrollo, lo que permite obtener la ubicación en el entorno de la plataforma de programación Android Studio.

Cuando se realiza un lanzamiento de este tipo de aplicación en la tienda Google (Play Store) es fundamental el modificar el código fuente o base del aplicativo en donde se integran la conexión con el encargado de seguridad. Además, en lo futuro es necesario realizar mejoras en el diseño e implementación de una aplicación móvil para alertar una emergencia de personas vulnerables sobre un teléfono inteligente basándose en el sistema operativo iOS.

REFERENCIAS

- Amnistía Internacional. (2014). La nueva aplicación “Botón de Pánico” ofrece una red de seguridad a los activistas de derechos humanos. Recuperado el 22 de Marzo de 2019 de <https://www.amnesty.org/es/latest/news/2014/05/new-panic-button-app-provides-safety-net-human-rights-activists/>
- AsT. (2018). Botón de Pánico. Recuperado el 23 de Marzo de 2019 de <http://www.botondepanicoast.com.ar>
- Báez, J., Borrego, M., & Cordero, A. (2016). Introducción al Android. Recuperado el 20 de Marzo de 2019, de <http://www.it-docs.net/ddata/18.pdf>
- Crockford, D. (s.f.). *Introducing JSON*. Recuperado el 25 de Marzo de 2019 de <http://www.json.org/>
- DiMarzio J.F. (2008). *Android Programming with Android Studio*. (4.^a ed.). Canada, United States of America: Jhon Wiley & Sons, Inc.
- Google LLC. (s.f.). *DB-Engines Ranking*. Recuperado el 30 de Marzo de 2019 de <https://db-engines.com/en/ranking>
- Google LLC. (2019). ¿Qué es streaming?. Recuperado el 22 de Abril de 2019 de <https://www.aboutespanol.com/que-es-streaming-157642>
- Google LLC. (2019). *BroadcastReceiver*. Recuperado el 30 de Marzo de 2019 de <https://developer.android.com/reference/android/content/BroadcastReceiver>
- Google LLC. (2019). Intents y filtros de intents. Recuperado el 05 de Abril de 2019 de <https://developer.android.com/guide/components/intents-filters>

- Google LLC. (2019). *Handling changes in audio output*. Recuperado el 05 de Abril de 2019 de <https://developer.android.com/guide/topics/media-apps/volume-and-earphones>
- Google LLC. (2019). *LocationListener*. Recuperado el 06 de Abril de 2019 de <https://developer.android.com/s/results/?q=location%20listener>
- Google LLC. (2019). *Actividades*. Recuperado el 01 de Abril de 2019 de <https://developer.android.com/guide/components/activities?hl=es-419>
- Google LLC. (2019). *Java Secure Socket Extension (JSSE)*. Recuperado el 01 de Abril de 2019 de <https://docs.oracle.com/javase/8/docs%2Ftechnotes%2Fguides%2Fsecurity%2Fjsse%2FJSSERefGuide.html>
- Google LLC. (2019). *MediaRecorder*. Recuperado el 02 de Junio 2019 de <https://developer.android.com/reference/android/media/MediaRecorder>
- INEC. (2008). *La población adulto mayor en la ciudad de Quito estudio de la situación sociodemográfica y socioeconómica*. Recuperado el 22 de Marzo de 2019 de http://www.ecuadorencifras.gob.ec/documentos/web-inec/Bibliotecas/Estudios/Estudios_Economicos/Evolucion_de_la_indus_Alimen_Beb_2001-2006/Poblac_Adulto_Mayor.pdf
- INEC. (2017). *Adulto mayor quienes son y cuántos son*. Recuperado el 22 de Marzo de 2019 de <http://igualdad.gob.ec/adulto-a-mayor/quienes-son-adulto-mayor.html>
- INNOVA. (2018). *Informes y tendencias*. Recuperado el 20 de Marzo de 2019 de http://www.seaceptanideas.com/docs/ticbeat%20%20desarrollo_de_aplicaciones_moviles.pdf
- Jan L. Harrington. (2016). *Relational Database Design and Implementation*. (4.^a ed.). San Francisco, CA, USA: Morgan Kaufmann Publishers, Inc.

Laudon & Laudon. (2012). Sistema de Información Gerencial. (12.^a ed.). Naucalpan de Juárez, Estado de México, México: Pearson Educación de México.

Merriam-Webster. (s.f.). *Definition of database*. Recuperado el 29 de Marzo de 2019 de <https://www.merriam-webster.com/dictionary/database>

Ministerio de Salud Pública (2019). Consejo Nacional para la igualdad de discapacidades: Personas con discapacidad registradas. Recuperado el 02 de Abril de 2019 de <https://www.consejodiscapacidades.gob.ec/>

Quora (s.f.). *What is firebase*. Recuperado el 30 de Marzo de 2019 de <https://www.quora.com/What-is-firebase>

Robledo, D. (s.f.) Desarrollo de aplicaciones para Android. (1.^a ed.). España: Ministerio de educación, cultura y deporte.

Tyco Integrated Fire & Security. (2013). Alarmas silenciosas: botón de pánico, alarmas anti-coacción y alarmas anti-atraco, Recuperado el 18 de Marzo de 2019 de <https://blogseguridad.tyco.es/productos/alarmas-silenciosas-boton-de-panico-alarmas-anti-coaccion-y-alarmas-anti-atraco/>

ANEXOS

Anexo 1 – Manual de Usuario

INTRODUCCIÓN

Este documento está orientado al usuario, dando a conocer todas y cada una de las funcionalidades de la aplicación móvil “Alerta de Emergencia”.

El objetivo de este manual es cumplir el uso correcto de las funcionalidades que posee la aplicación móvil en el cual se dispone de descripciones detalladas:

1. Instalar apk

Activar las opciones de desarrollador haciendo varios clicks en la opción Compilación de “Acerca del teléfono”.



Figura 1. Opciones de desarrollador.

Instalar el apk de la siguiente manera:

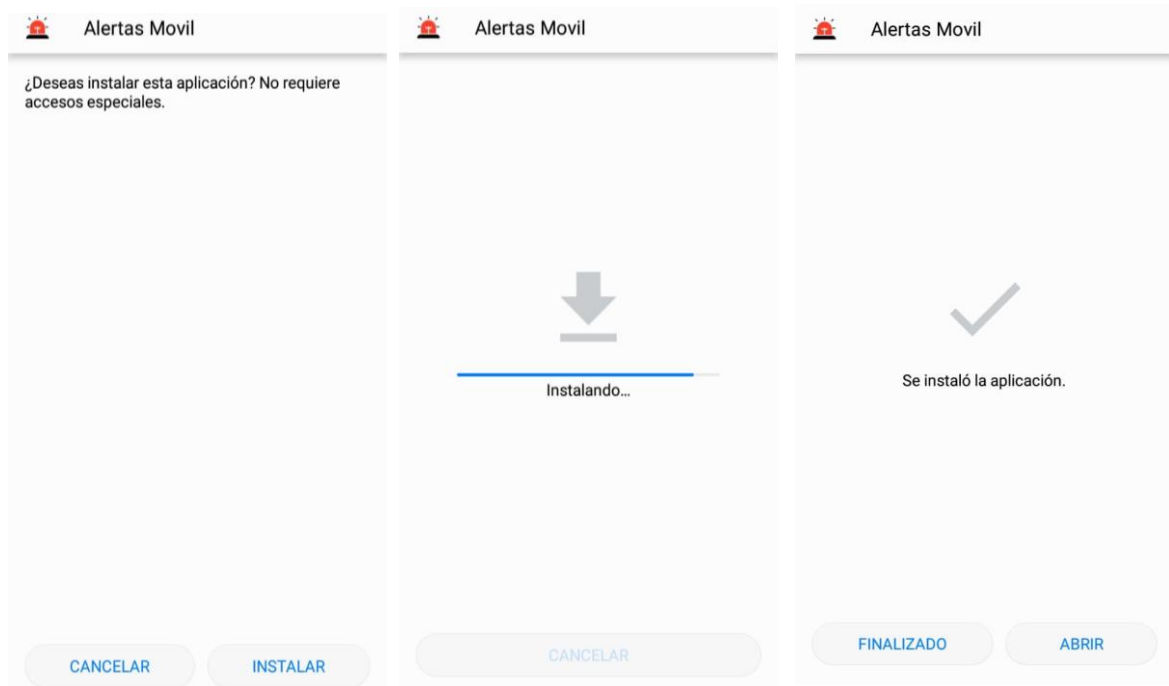


Figura 2. Instalación del archivo apk.

2. Al instalar la aplicación se mostrará una pantalla de inicio



Figura 3. Ícono de la aplicación.

3. Inicio de sesión en la aplicación Android

Ésta se abre y se muestra el correo del usuario de modo que se despliega la pantalla que corresponde a la pantalla de bienvenida al usuario.

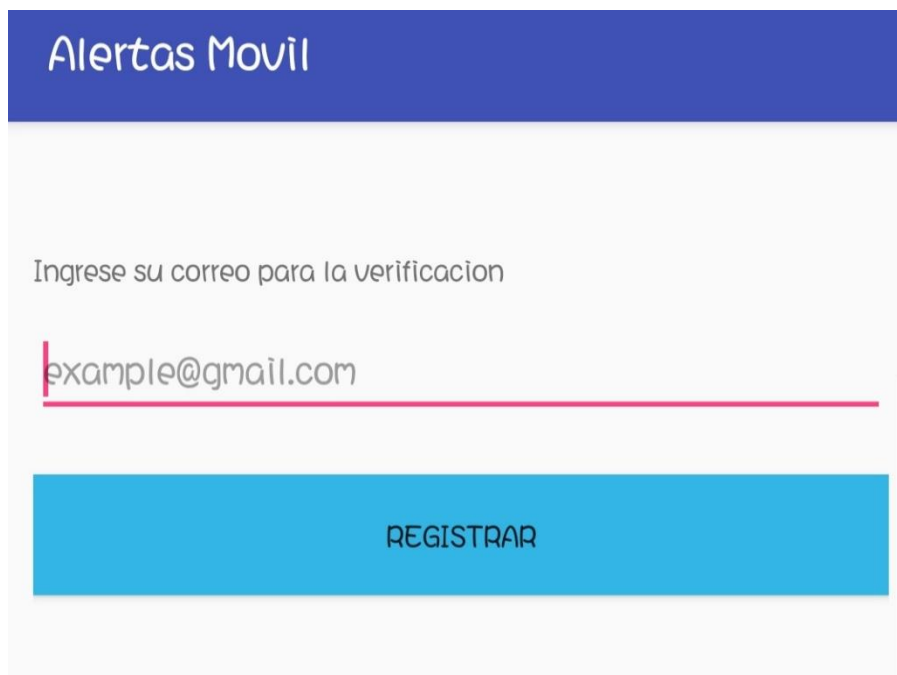
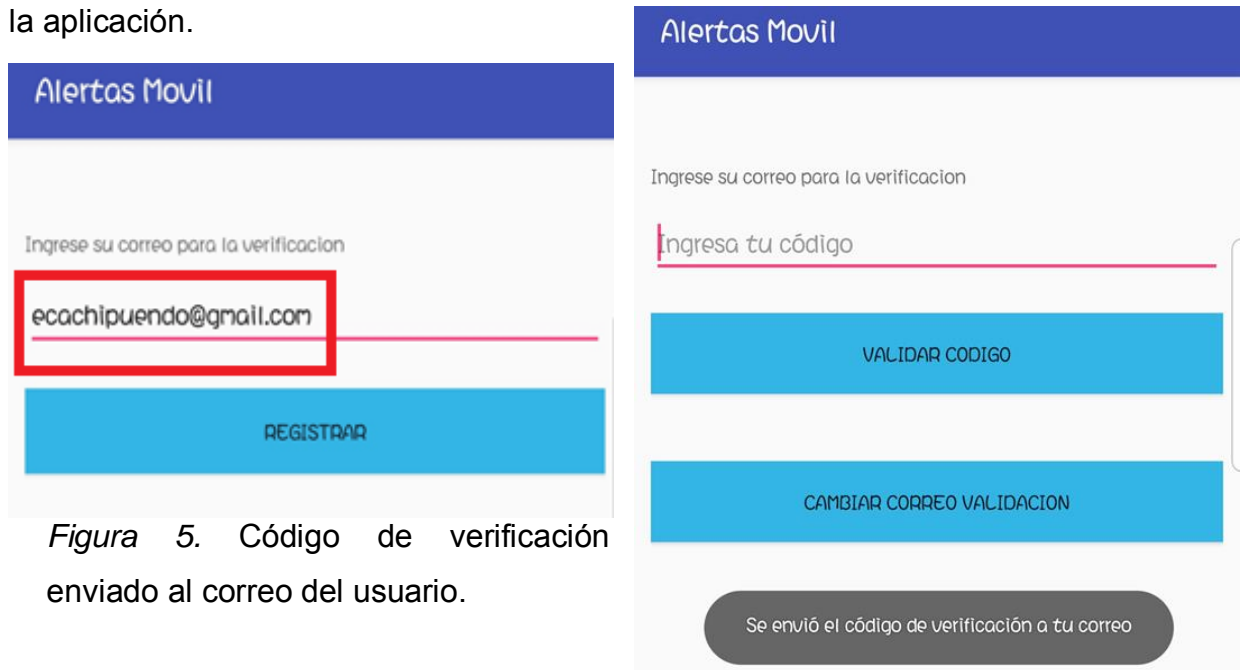


Figura 4. Registro e inicio de sesión.

4. Código de verificación

Se ingresa un correo válido al cual le llegará un código de 5 dígitos para validar la aplicación.



Alertas Movil

Ingrese su correo para la verificacion

ecachipuendo@gmail.com

REGISTRAR

Alertas Movil

Ingrese su correo para la verificacion

Ingres tu código

VALIDAR CODIGO

CAMBIAR CORREO VALIDACION

Se envió el código de verificación a tu correo

Figura 5. Código de verificación enviado al correo del usuario.

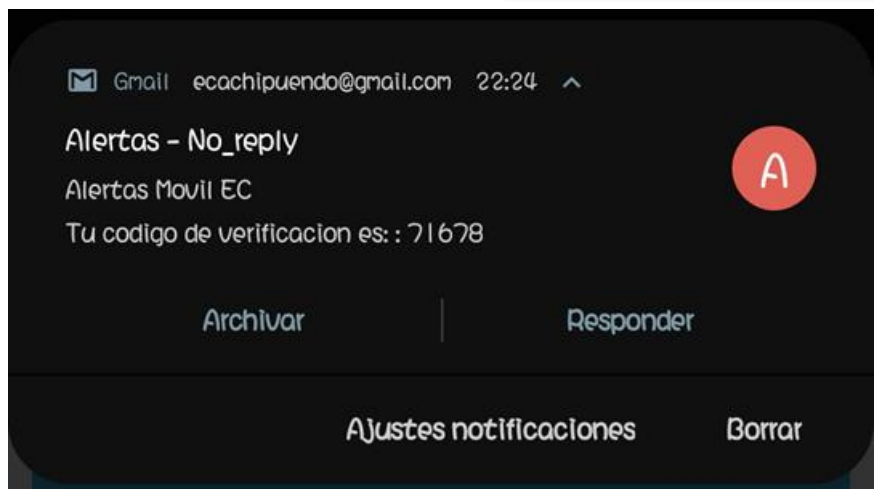


Figura 6. Código de verificación.

Los usuarios podrán acceder a la aplicación, una vez que se haya registrado y validado el código de verificación a su correo.

Una vez ingresado el código de verificación, se muestra la pantalla principal con las siguientes opciones:



On: Para iniciar el servicio de la aplicación

Off: Para detener el servicio de la aplicación

Mi Ubicación: Para obtener información basada en su ubicación

Perfil: Para ingresar/verificar los datos del usuario

Figura 7. Pantalla de inicio.

5. Opción “Perfil”

Datos del usuario

Datos del contacto de emergencia para poder enviar la alerta

Duración del video de alerta

Figura 8. Registro de datos del usuario.

6. Enviar Alerta



Figura 9. Envío alerta al servidor y a la persona de contacto.

