



**UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MÉXICO**  
**FACULTAD DE INGENIERÍA**

*DESARROLLO DE UNA APLICACIÓN MÓVIL CON INTERFAZ  
GESTUAL PARA EL ENVÍO DE ALARMAS ANTE SITUACIONES DE  
RIESGO*

# TESIS

QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN

PRESENTA:

José Miguel Alonso Muñoz

ASESOR:

Dr. Marcelo Romero Huertas



TOLUCA MÉXICO, NOVIEMBRE 2020

## RESUMEN

Lamentablemente, la inseguridad en la vía pública en nuestros días ha alcanzado índices altos, siendo los robos uno de los incidentes más comunes; los objetos más robados: carteras, relojes, equipos de cómputo portátiles y teléfonos celulares.

Generalmente, un asaltante es rápido e intimida al asaltado para despojarlo de sus bienes, evitando llamar la atención de los transeúntes y de que éste pida ayuda. El teléfono celular es un dispositivo de comunicación portátil y que está en operación continua a pesar de tener bloqueada su pantalla, por lo que, generalmente es el primer objeto hurtado.

Por otro lado, las generaciones recientes de teléfonos celulares están diseñados con una serie de sensores, los cuales permiten monitorear el entorno y hacen posible la activación de eventos con base en la información capturada. Por consiguiente, dichos sensores pueden utilizarse para enviar notificaciones a través de una interfaz no convencional, podría ser útil en una situación de robo.

Esta tesis propone una aplicación que a través de una interfaz gestual envíe una notificación y la ubicación registrada por el teléfono celular, para enterar a un contacto que el dueño del celular se encuentra en una situación de riesgo.

En el trabajo, se presenta una aplicación móvil para la plataforma operativa Android de las versiones 5.0 a la 9.0, la cual, contraria a otras plataformas operativas, no restringe el uso de los sensores del teléfono celular. Dicha aplicación móvil, está codificada con los lenguajes de programación *Java 1.8* y *Extensible Markup Language (XML)* y tiene como gestor de base de datos *SQL SERVER 2012*. Es importante destacar que, esta aplicación está diseñada para teléfonos inteligentes, los cuales generalmente cuentan con un acelerómetro como sensor de movimiento.

Esta aplicación móvil está diseñada con tres funcionalidades principales: instalación, configuración y notificación. La instalación hace uso de *Google Drive* como servidor de descarga, ya que éste cuenta con acceso público y permite que cada usuario descargue e instale la aplicación desde su teléfono celular. La configuración consiste en el registro de datos personales del usuario, tales como nombre y correo electrónico; adicionalmente, el usuario define los datos de su contacto de emergencia, incluyendo su nombre y número de celular, así como el mensaje de notificación a ser enviado. Finalmente, la notificación consiste en el envío de un mensaje de alarma al contacto de emergencia registrado.

La característica técnica más importante de la aplicación móvil propuesta en esta tesis, es el envío del mensaje de alarma a través de una interfaz gestual, la cual se activa con un movimiento natural de la mano del usuario. Este movimiento de mano consiste en realizar un giro de 180° cuidando que la pantalla del teléfono celular apunte hacia el suelo al momento de ser entregado al agresor. Entonces, dicho movimiento es leído y codificado por el acelerómetro del dispositivo y envía al contacto de emergencia un mensaje SMS con el texto configurado, además de un hipervínculo de la aplicación *Google Maps* activada con las coordenadas geográficas de localización del usuario. Éstas coordenadas geográficas son la latitud y longitud detectadas automáticamente por el GPS del teléfono celular.

La evaluación de la aplicación móvil propuesta en esta tesis, se realiza en la vía pública de tres colonias del Estado de México: Santa Cruz Atzacapotzaltongo y Nueva Oxtotitlán del municipio de Toluca, y El Calvario del municipio de Zinacantepec, las cuales se distinguen por su índice delictivo medio alto.

Las tareas definidas son evaluadas con dos métricas de usabilidad de rendimiento y una de preferencia: Total de operaciones realizadas correctamente, tiempo que toma en realizar la tarea asignada, facilidad de uso de la aplicación.

Dentro de los resultados obtenidos en la evaluación de la aplicación móvil se distingue lo siguiente. Para la tarea *instalación de la aplicación* se obtiene un tiempo medio de 37.2 segundos. En cuanto a la tarea *configuración de la aplicación* se obtiene un tiempo medio de 48.43 segundos. Con respecto a la tarea *envío de la notificación* se obtiene un número de operaciones realizadas correctamente del 100% y con 12 segundos de tiempo medio. Finalmente, la tarea *registro de la alarma* se obtiene un tiempo medio de 12 segundos.

## TABLA DE CONTENIDO

Dedicatorias .....	3
Resumen .....	4
Introducción.....	8
Objetivo general .....	9
Hipótesis .....	9
Alcances.....	9
Organización del documento .....	9
Capítulo I. Generalidades.....	10
1.1 Dispositivos móviles .....	10
1.1.1 Sensores en dispositivos móviles .....	10
1.2 Sistema operativo para dispositivos móviles .....	12
1.2.1 Sistema Operativo Android .....	13
1.2.2 Arquitectura del sistema operativo Android.....	13
1.2.3 Sistema Operativo iOS.....	15
1.3 Herramientas Tecnológicas .....	16
1.3.1 Programación Orientada a Objetos.....	16
1.3.2 JAVA.....	16
1.3.3 XML.....	17
1.3.4 Android studio .....	17
1.3.5 Arquitectura de servicios .....	18
1.4 Interacción hombre máquina.....	18
1.4.2 Estructura de la interacción Hombre – Máquina.....	19
1.4.3 Objetivos de la Interacción Humano-Computadora .....	20
1.4.4 Diseño.....	21
1.5 Metodología de desarrollo centrada en el usuario .....	22
1.6 Naturaleza de las Interfaces Gestuales .....	25
1.6.1 Gesto.....	25
1.6.2 Tipos de gestos.....	26
1.6.3 Clasificación por dispositivos de entrada .....	27
1.6.4 Consideraciones de diseño de una interfaz gestual .....	28
Capítulo II. Desarrollo de la aplicación móvil.....	31
2.1 Análisis .....	31
2.2 Métricas de usabilidad .....	33
2.3 Diseño .....	34

2.4 Prototipo funcional.....	49
2.4.1 Prototipo inicial.....	49
2.4.2 Primer refinamiento.....	50
Capítulo III. Evaluación de la aplicación móvil.....	57
3.1 Evaluación del prototipo inicial.....	57
3.2 Evaluación del primer refinamiento del prototipo.....	58
3.2.1 Pruebas basadas en el usuario.....	58
3.2.1.1 Selección de usuarios.....	58
3.2.1.2 Tareas a realizar.....	61
3.2.1.3 Métricas de usabilidad.....	69
3.2.1.4 Escenario de experimentación.....	69
3.2.1.5 Proceso de evaluación.....	70
3.3 Resultados.....	72
3.3.1 Tareas y resultados de la evaluación.....	72
Conclusiones y trabajo futuro.....	76
Anexo A. Encuesta.....	77
Referencias.....	78

## INTRODUCCIÓN

---

Actualmente ha aumentado la cantidad de robos y asaltos a mano armada en ciertos puntos del Estado de México, tanto con peatones en las vías públicas como en el transporte público, que hacen necesario encontrar una manera factible de notificar a las autoridades de una forma discreta que no comprometa la seguridad del ciudadano afectado.

Los desarrollos actuales de notificaciones emplean un paradigma de interacción estándar [1-5], generalmente basado en interfaces gráficas de usuario con botones, menús o táctiles. Lamentablemente, cuando un individuo se enfrenta a un asalto o está comprometida su seguridad, dicho paradigma de interacción no es práctico, puesto que el individuo experimenta nerviosismo al ser condicionado por el delincuente. Por lo que, esta tesis desarrolla una aplicación que envíe dichas notificaciones utilizando una interfaz gestual (paradigma de interacción no estándar) [6-7] basada en gestos corporales, tal como el movimiento de manos que discretamente envíen una notificación a un contacto en particular para que dicho contacto tenga noción de la situación, así como también del lugar en el cual están sucediendo los hechos.

Actualmente no existen trabajos de investigación que resuelvan este tipo de problemática, sin embargo, un trabajo relacionado es [11], donde se analizan las actividades cotidianas hechas por determinado número de usuarios y recopilan los datos a través de los dispositivos móviles de los usuarios. En ese trabajo el análisis de movimiento se realiza con base en la información proporcionada por acelerómetros.

Como se ha mencionado, uno de los problemas que vive actualmente la sociedad son los asaltos en el transporte público y en las vialidades. Por ejemplo, las estadísticas del INEGI [10] en el 2019 indican que, de cada 4736 denuncias mensuales, el 30% es por robos y/o asaltos a la ciudadanía.

Ante esas situaciones, la mayoría de las personas genera un estrés elevado, provocándole miedo que los condiciona a seguir literalmente las indicaciones del asaltante.

Como bien se sabe, actualmente los dispositivos móviles cuentan con diversos componentes que le permiten comunicarse con otros dispositivos, generalmente con los protocolos Bluetooth o Wifi. Lo que permite habilitar aplicaciones de uso específico que comuniquen a los usuarios en redes locales y abiertas.

Las interfaces de acceso estándar son interfaces que requieren el uso del teclado (físico o táctil), pero durante un asalto no son prácticas porque el usuario en esa situación tiene poco control de sus acciones y poco tiempo, debido al estrés en el que se encuentra.

El desarrollo de este trabajo podría brindar apoyo a la ciudadanía para hacer notificación a las autoridades y estos a su vez puedan establecer las medidas requeridas para controlar y prevenir robos y asaltos.

Por lo que se realiza la creación de un primer prototipo que permita el envío de alarmas para aviso a un contacto personalizado, considerando que el empleo de las interfaces de usuario de acceso estándar no son prácticas cuando el individuo está siendo asaltado y se encuentra comprometida su seguridad.

Debido a ello, se propone el uso de interfaces de usuario no estándar, tal como interfaces gestuales basadas en el movimiento corporal y de las manos. Para esto, se debe analizar la arquitectura hardware de los dispositivos móviles, que permitan realizar esta tarea, ej. sensores de movimiento.

Finalmente, se hace notar que el desarrollo realizado en esta tesis tiene diversas áreas de aplicación, como la seguridad, la industria, la medicina y la rehabilitación, porque este desarrollo puede ser enfocado para

realizar notificaciones de diferente tipo no solamente para reportar robos y asaltos a mano armada, ampliando el panorama para poder notificar a diferentes personas de acuerdo al caso, por ejemplo, que una persona en tratamiento clínico pueda enviar una notificación al médico tratante sobre alguna situación de emergencia.

### OBJETIVO GENERAL

Desarrollar una aplicación móvil con interfaz gestual que permita leer y codificar los movimientos de mano para el envío de alarmas cuando el usuario enfrente situaciones de riesgo.

### HIPÓTESIS

Es posible enviar un 80% de alarmas verdaderas-positivas utilizando una aplicación móvil habilitada con interfaz gestual cuando el usuario enfrenta situaciones de riesgo.

### ALCANCES

Se presenta una aplicación funcional que envía mensajes de texto a un contacto por medio de un gesto o comando de interfaz gestual, para notificar que el usuario se encuentra en peligro y en donde se está llevando a cabo el suceso.

La aplicación está diseñada para que el sensor de movimiento envíe una notificación con la ubicación de la persona al momento que se realice el gesto de comando designado. Sin embargo, esta aplicación no es capaz de identificar si el usuario la activó de manera inconsciente.

Por otro lado, el envío de la alarma está comprometido ante las circunstancias en las que se presenta un robo. Por ejemplo, un caso es cuando el asaltante sea quien despoje al usuario de su teléfono celular y por tanto el usuario no podría ejecutar el gesto que active la alarma.

Un requerimiento técnico de la aplicación móvil propuesta en esta tesis, es que necesita la disponibilidad de la red de comunicaciones de una compañía celular para el envío de una notificación.

En el mismo sentido, la alarma se envía siempre y cuando se ejecute el comando de movimiento correcto y en la posición correcta.

### ORGANIZACIÓN DEL DOCUMENTO

La presente tesis está integrada por cuatro capítulos:

El Capítulo I – “Generalidades”, describe las generalidades de los dispositivos móviles viables para las aplicaciones Android.

El Capítulo II – “Análisis y desarrollo de la interfaz gestual”, documenta el diseño e implementación de una aplicación Android para el envío de alarmas.

El Capítulo III – “Evaluación y análisis de resultados”, describe el proceso de evaluación y analiza los resultados obtenidos.

# CAPÍTULO I. GENERALIDADES

---

*En este capítulo se describen los conceptos básicos para que pueda ser entendible la información que se mostrará en los siguientes capítulos.*

## 1.1 DISPOSITIVOS MÓVILES

Actualmente existen diversos dispositivos móviles, en esta sección se describen algunos de ellos.

*Smart Phones* (Teléfonos Inteligentes): Es un dispositivo construido sobre una plataforma informática móvil, con capacidad de almacenar datos y realizar tareas similares a las de una computadora de escritorio o laptop.

Estos dispositivos cuentan con una pantalla táctil, pero siguen conservando las características de un teléfono celular. Dichas funciones son envío y/o recepción de mensajes, llamadas telefónicas, etc. [28].

*Tablets* (Tabletas): Es un computador portátil de mayor tamaño en comparación a un teléfono móvil, sus características distintivas son: sus capacidades para conexión con otros dispositivos como proyector, etc. [30].

*Smart Watch* (Reloj Inteligente): Es un reloj de pulsera dotado con funcionalidades como: acceso a internet, realizar y recibir llamadas telefónicas, enviar y recibir emails y recibir notificaciones del *Smart Phone* e incluso consultar las redes sociales [29].

Consola de videojuegos portátil, son dispositivos diseñados especialmente para jugar, adicionalmente se puede escuchar música, almacenar información como fotos o archivos, navegar por Internet, se diferencia de las consolas normales, ya que son más ligeras y fáciles de transportar.

*Laptop*, es una computadora portátil de peso y tamaño ligero, su tamaño es aproximadamente el de un portafolio, esta pertenece al grupo de las computadoras personales, los cuales son equipos de cómputo relativamente pequeños y de bajo costo.

*E-Book*, son un tipo de dispositivo portátil que permite almacenar y leer libros digitalizados, o cualquier otro tipo de documento escrito que contenga imágenes, el objetivo de los E-Book es proveer una alternativa a los periódicos y libros tradicionales que están impresos en papel.

### 1.1.1 SENSORES EN DISPOSITIVOS MÓVILES

Entre los sensores convencionales en los dispositivos móviles actuales se tiene: acelerómetro, barómetro, giroscopio, magnetómetro, proximidad, luz y GPS. En la Tabla 1.1 se da una breve descripción de su funcionalidad en los dispositivos móviles [32].

TABLA 1.1 SENSORES EN MÓVILES

Sensor	¿Qué es?	Funcionalidad
Acelerómetro	El acelerómetro es un componente mecánico muy parecido a un chip, de un tamaño reducido gracias a su nanotecnología, y fabricado en silicio.	El acelerómetro sirve para que el teléfono celular sepa en qué orientación está colocado, de manera que el dispositivo sabe cuándo se encuentra de forma horizontal, o en vertical, e incluso cuándo lo has colocado boca abajo.
Barómetro	Se trata de un sensor y su propósito es la de medir la presión ambiental del aire.	Con esto, el teléfono celular inteligente puede medir la altura a la que se encuentra, lo que en algunas aplicaciones puede ayudar con el posicionamiento del GPS.
Giroscopio	El giroscopio es otro sensor que mide la aceleración no gravitacional, y que está destinado a complementar la información sobre la orientación del móvil que ofrece el acelerómetro.	Para eso, añade una cuarta dimensión de movimiento que mide la rotación o el giro del móvil.
Magnetómetro	Es un sensor electrónico que miden y cuantifican las fuerzas magnéticas.	Suele utilizarse como brújula electrónica configurándolo para detectar el polo norte magnético de la tierra, y así definir dónde está el polo geográfico.
Proximidad	Está compuesto por un LED infrarrojo que emite un rayo invisible al ojo humano, y un receptor de infrarrojos que detecta la vuelta del rayo cuando rebota con una superficie.	Es el encargado de permitirle al teléfono celular saber, por ejemplo, cuándo acercamos el teléfono celular a la cara para que este pueda apagar la pantalla.
Luz	El sensor de luz tiene como propósito detectar la cantidad de luz que hay en el ambiente.	Con ello, el teléfono celular es capaz de gestionar el brillo de la pantalla cuando se tiene activada la opción de brillo automático.
GPS	Los sensores GPS del teléfono celular están diseñados para estar continuamente leyendo la señal que le transmiten los satélites de GPS, utilizando por lo general la señal de varios de ellos para triangular tu posición.	Por lo general, suele utilizar los ángulos de intersección de al menos tres satélites respecto a nuestro teléfono celular para triangular dónde se encuentra.

## 1.2 SISTEMA OPERATIVO PARA DISPOSITIVOS MÓVILES

Un sistema operativo es el conjunto de programas informáticos que permite la administración eficaz de los recursos de una computadora, es conocido como sistema operativo o software de sistema. Estos programas comienzan a trabajar apenas se enciende el equipo; ya que gestiona el hardware desde los niveles más básicos y permiten además la interacción con el usuario.

Cabe destacar que los sistemas operativos no sólo funcionan en las computadoras. Por el contrario, este tipo de sistema se encuentran en la mayoría de los dispositivos electrónicos que utilizan microprocesadores: el software de sistema posibilita que el aparato cumpla con sus funciones.

El sistema operativo cumple con cinco actividades principales: el suministro de interfaz de usuario, administración de recursos, administración de archivos, administración de tareas y servicio de soporte y utilidades [35].

En cuanto a la interfaz de usuario, el sistema se encarga de que el sujeto pueda acceder a archivos y realizar otras tareas con la computadora. La administración de recursos permite el manejo del hardware, incluyendo los periféricos y la red. El software de sistema también se encarga de la gestión de archivos, al controlar la creación, eliminación y acceso a los mismos, y de la administración de las tareas informáticas.

Por último, se puede mencionar que el servicio de soporte se encarga de actualizar las versiones, mejorar la seguridad del sistema, agregar nuevas utilidades, controlar y corregir los errores del software.

Un sistema operativo móvil es un conjunto de programas de bajo nivel que permite la abstracción de las peculiaridades del *hardware* específico del teléfono móvil y provee servicios a las aplicaciones móviles, que se ejecutan sobre él. Al igual que los PCs que utilizan *Windows* o *Linux* o *Mac OS*, los dispositivos móviles tienen sus sistemas operativos como *Android*, *IOS* *Windows Phone*, entre otros.

Los sistemas operativos móviles son mucho más simples y están más orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información en ellos.

Algunos de los sistemas operativos utilizados en dispositivos móviles están basados en el siguiente modelo de capas:

- *Kernel*: *Kernel* o núcleo es el programa que tiene el control sobre los componentes de hardware del dispositivo. También ofrece servicios a las capas superiores como son los controladores para el manejo del hardware, gestión de procesos, control sobre el sistema de archivos y acceso y gestión de la memoria.
- *Middleware*: Es el conjunto de módulos que hacen posible la propia existencia de aplicaciones para móviles. Es totalmente transparente para el usuario y ofrece servicios claves como, por ejemplo: motor de mensajería y comunicaciones, códecs multimedia, intérpretes de páginas web.
- Entorno de ejecución de aplicaciones: Consiste en un gestor de aplicaciones y un conjunto de interfaces de programación de aplicaciones que pueden ser programables abiertas y programables por parte de los desarrolladores para facilitar la creación de software.
- Interfaz de usuario: Facilitan la interacción con el usuario y el diseño de la presentación visual de la aplicación. Los servicios que incluye son el de componentes gráficos (botones, pantallas, listas, etc.) y el del marco de interacción

Aparte de estas capas también existe una familia de aplicaciones nativas del teléfono que suelen incluir los menús, el marcador de números, etc.

A continuación, se mencionará algunos sistemas operativos más comerciales, así como también de sus características y funciones principales.

### 1.2.1 SISTEMA OPERATIVO ANDROID

Es un sistema operativo para móviles desarrollado por *Google*, basado en el *kernel* de *Linux* y diseñado principalmente para la pantalla táctil dispositivos móviles como teléfonos inteligentes y tabletas.

La interfaz de usuario de *Android* se basa principalmente en la manipulación directa, usando gestos táctiles que corresponden vagamente a las acciones del mundo real, tales como deslizar, tocar, etc., para manipular objetos en la pantalla, junto con un teclado virtual para introducir texto.

El código fuente de *Android* fue liberado por *Google* bajo una licencia de código abierto, aunque la mayoría de los dispositivos *Android* en última instancia se envían con una combinación de código libre y propietario de *software*, incluyendo *software* propietario requerido para acceder a los servicios de *Google*.

*Android* es muy popular entre las empresas de tecnología que requieren un sistema operativo personalizable para alta tecnología en dispositivos. Su carácter abierto ha alentado una gran comunidad de desarrolladores entusiastas para utilizar el código fuente abierto como base para proyectos impulsados por la comunidad, que proporcionan actualizaciones de los dispositivos más antiguos, añadir nuevas funciones para usuarios avanzados o traer *Android* para dispositivos vendidos originalmente con otros sistemas operativos.

La amplia variación de *hardware* de los dispositivos *Android* provoca retrasos significativos para las actualizaciones de *software*, con las nuevas versiones del sistema operativo y los parches de seguridad típicamente toman meses antes de llegar a los consumidores.

### 1.2.2 ARQUITECTURA DEL SISTEMA OPERATIVO ANDROID

En esta sección, se da una breve explicación del sistema operativo *Android*, así como de los elementos que componen dicho sistema operativo.

En la Figura 1.1 se muestra la arquitectura con la que cuenta un sistema operativo *Android* [33]. Adicionalmente, se da una breve explicación de cada capa que compone el sistema operativo *Android*.

- a) Aplicaciones, La capa superior de esta pila de *software* la forman, las aplicaciones. En este caso se incluyen todas las aplicaciones del dispositivo, tanto las que tienen interfaz de usuario como las que no, tanto las nativas (programadas en C o C++) como las administradas (programadas en *Java*), tanto las que vienen de serie con el dispositivo como las instaladas por el usuario.
- b) Marco de Aplicación está formado por todas las clases y servicios que utilizan directamente las aplicaciones para realizar sus funciones y que, obviamente, se apoyan en las bibliotecas y en el entorno de ejecución que ya hemos detallado. La mayoría de los componentes de esta capa son bibliotecas *Java* que acceden a los recursos a través de la máquina virtual *Dalvik*.
- c) Bibliotecas, la componen las bibliotecas nativas de *Android* (también mal llamadas librerías, por el término inglés *libraries*). Estas bibliotecas están escritas en C o C++ y compiladas para la arquitectura *hardware* específica del teléfono, tarea que normalmente realiza el fabricante, que también se encarga de instalarlas en la terminal antes de ponerlo a la venta. Su cometido es

proporcionar funcionalidad a las aplicaciones, para tareas que se repiten con frecuencia, evitando tener que codificarlas cada vez y garantizando que se llevan a cabo de la forma más eficiente.

- d) El entorno de ejecución se apoya en las bibliotecas, no se considera una capa en sí mismo, dado que también está formado por bibliotecas. En concreto, las bibliotecas esenciales de Android incluyen la mayoría de la funcionalidad de las bibliotecas de *Java*, así como otras específicas de Android. Los componentes que forman el entorno de ejecución son:
- a. Máquina Virtual *Dalvik* es la máquina virtual que utiliza la plataforma para dispositivos móviles Android, la cual permite ejecutar aplicaciones programadas en *Java*.
  - b. *Runtime* es un conjunto de bibliotecas base que proporcionan la mayor parte de las funciones disponibles en las bibliotecas base del lenguaje *Java*. Cada aplicación *Android* corre su propio proceso, con su propia instancia de la máquina virtual *Dalvik*.

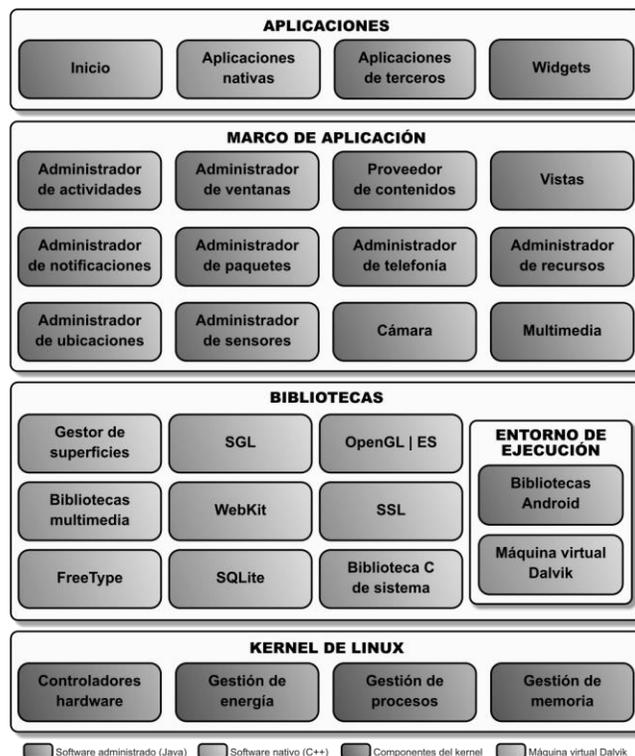


FIGURA 1.1 ARQUITECTURA ANDROID [33].

- e) *Kernel de Linux*, es el núcleo del sistema operativo *Android* es un *kernel Linux* versión 2.6, similar al que puede incluir cualquier distribución de Linux, como Ubuntu, solo que adaptado a las características del hardware en el que se ejecutará *Android*.
- a. Proporciona una capa de abstracción para los elementos *hardware* a los que tienen que acceder las aplicaciones. Esto permite que se pueda acceder a esos componentes sin necesidad de conocer el modelo o características precisas de los que están instalados en cada teléfono. De esta forma, si una aplicación necesita, por ejemplo, la brújula, podrá utilizar la que incluya el teléfono, sea cual sea. Para cada elemento hardware del teléfono existe un controlador dentro del *kernel* que permite utilizarlo desde el software.

- b. Además de proporcionar controladores *hardware*, el *kernel* se encarga de gestionar los diferentes recursos del teléfono (energía, memoria, etc.) y del sistema operativo en sí procesos, elementos de comunicación, etc.

### 1.2.3 SISTEMA OPERATIVO IOS

Es un sistema operativo para móviles de la multinacional *Apple Inc.* Originalmente desarrollado para el *iPhone (iPhone OS)*, después se ha usado en dispositivos como el *iPod touch* y el *iPad*. No permite la instalación de *iOS* en *hardware* de terceros.

Los elementos de control consisten en deslizadores, interruptores y botones. La respuesta a las órdenes del usuario es inmediata y provee una interfaz fluida. La interacción con el sistema operativo incluye gestos como deslices, toques, los cuales tienen definiciones diferentes dependiendo del contexto de la interfaz. Se utilizan acelerómetros internos para hacer que algunas aplicaciones respondan a sacudir el dispositivo (por ejemplo, para el comando deshacer) o rotarlo en tres dimensiones (un resultado común es cambiar de modo vertical a horizontal).

En la Figura 1.2 se muestra la arquitectura con la que cuenta un sistema operativo *iOS* [34].

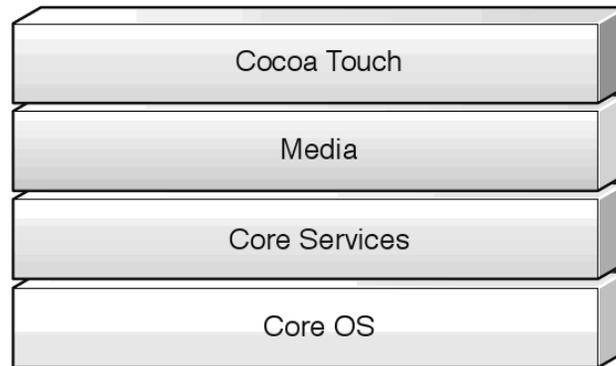


FIGURA 1.2 ARQUITECTURA DE IOS [34]

A continuación, se describe brevemente cada capa que compone la arquitectura del sistema operativo *iOS*.

- a) *Cocoa Touch* o capa táctil de Cocoa es la interfaz de los dispositivos para con el usuario. Esta capa es una exclusiva interfaz con un diseño único que está constituida con aplicaciones probadas que comparten muchos patrones que se encuentran en la *PC Mac*, pero se reconstruyeron con un enfoque especial en el tacto y las interfaces basadas en optimización. La mayoría de estas aplicaciones están basadas en *Objective C*.
- b) Capa Media o también como capa de “medios de comunicaciones”. Es una interfaz basada en una mezcla de lenguaje C y *objective C* que permite la ejecución de tareas o gestión de ficheros multimedia. Esta capa contiene las tecnologías de gráficos, audio y video orientadas a crear la mejor experiencia de multimedia disponible en un dispositivo móvil. Las tecnologías están diseñadas para hacer que las aplicaciones se vean y suenen excelente.
- c) *Core Services*, esta capa contiene los servicios fundamentales del sistema para las aplicaciones. La llave entre estos servicios son el fundamento. Esta capa también contiene tecnologías individuales para admitir características como la ubicación, iCloud, medios de comunicación social, y la creación de redes.

- d) *Core OS*, en esta capa se incluyen las características de bajo nivel en que se basan la mayoría de otras tecnologías. Incluso si usted no utiliza estas tecnologías directamente en sus aplicaciones. Y en situaciones donde se necesita para hacer frente de forma explícita con la seguridad o la comunicación con un accesorio de hardware externo, lo hace utilizando el marco de esta capa.

## 1.3 HERRAMIENTAS TECNOLÓGICAS

En esta sección se describen diversos elementos de programación, estos elementos serán empleados secciones posteriores a esta.

### 1.3.1 PROGRAMACIÓN ORIENTADA A OBJETOS

La programación orientada a objetos Es un paradigma de programación que usa objetos y sus interacciones, para diseñar *software*. Está basado en cuatro pilares principalmente los cuales son:

- **Abstracción:** es la capacidad de obtener y aislar toda la información y cualidades de un objeto que no nos parezcan relevantes, para poder encapsularlos.
- **Herencia:** es la capacidad de crear un objeto a partir de otro objeto ya existente. El nuevo objeto hereda todas las cualidades del objeto del que deriva y además puede añadir nuevas funcionalidades o modificar las ya existentes.
- **Polimorfismo:** se refiere a que el mismo método puede ser usado para diferentes fines según se necesite.
- **Encapsulación:** es la capacidad de ocultar los datos abstraídos, aislarlos o protegerlos de quién no desees que tenga acceso a ellos.

Ventajas:

- Reutilización de los componentes.
- Facilidad de mantenimiento
- Modificación de los objetos existentes.
- Estructuración modular clara.
- Facilidad de creación de interfaz gráfica de usuario (*GUI*).
- Alto acoplamiento con bases de datos.

Desventajas

- **Limitaciones del programador:** Es posible que el programador desconozca algunas características del paradigma y de hecho siga utilizando el paradigma estructurado.
- No hay una forma única de resolver los problemas. Esto puede llevar a que diferentes interpretaciones de la solución planteada emerjan.

### 1.3.2 JAVA

*Java* es un lenguaje de programación orientado a objetos, puro, concurrente, que fue diseñado específicamente para tener tan pocas dependencias de implementación como fuera posible. Su intención es permitir que los desarrolladores de aplicaciones escriban el programa una vez y lo ejecuten en cualquier dispositivo (conocido en inglés como *WORA*, o "*write once, run anywhere*"), lo que quiere decir que el código que es ejecutado en una plataforma no tiene que ser recompilado para correr en otra. Java es, a partir de 2012, uno de los lenguajes de programación más populares en uso, particularmente para aplicaciones de cliente-servidor de web, con unos diez millones de usuarios reportados.

El lenguaje de programación Java fue originalmente desarrollado por *James Gosling*, de *Sun Microsystems* (constituida en 1982 y posteriormente adquirida el 27 de enero de 2010 por la compañía *Oracle*), y publicado en 1995 como un componente fundamental de la plataforma Java de *Sun Microsystems*. Su sintaxis deriva en gran medida de C y C++, pero tiene menos utilidades de bajo nivel que cualquiera de ellos. Las aplicaciones de Java son compiladas a *bytecode* (clase *Java*), que puede ejecutarse en cualquier máquina virtual Java (*JVM*) sin importar la arquitectura de la computadora subyacente.

La compañía *Sun* desarrolló la implementación de referencia original para los compiladores de Java, máquinas virtuales y librerías de clases en 1991, y las publicó por primera vez en 1995. A partir de mayo de 2007, en cumplimiento de las especificaciones del Proceso de la Comunidad Java, *Sun* volvió a licenciar la mayoría de sus tecnologías de Java bajo la Licencia Pública General de *GNU*. Otros también han desarrollado implementaciones alternas a estas tecnologías de *Sun*, tales como el Compilador de *Java* de *GNU* y el *GNU Classpath*.

### 1.3.3 XML

*XML*, siglas en inglés de *eXtensible Markup Language*, traducido como "Lenguaje de Mercado Extensible" o "Lenguaje de Marcas Extensible", es un metalenguaje que permite definir lenguajes de marcas desarrollado por el *World Wide Web Consortium (W3C)* utilizado para almacenar datos en forma legible. Proviene del lenguaje *SGML* y permite definir la gramática de lenguajes específicos (de la misma manera que *HTML* es a su vez un lenguaje definido por *SGML*) para estructurar documentos grandes. A diferencia de otros lenguajes, *XML* da soporte a bases de datos, siendo útil cuando varias aplicaciones deben comunicarse entre sí o integrar información.

*XML* no ha nacido únicamente para su aplicación en Internet, sino que se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Se puede usar en bases de datos, editores de texto, hojas de cálculo y casi cualquier cosa imaginable.

*XML* es una tecnología sencilla que tiene a su alrededor otras que la complementan y la hacen mucho más grande, con unas posibilidades mucho mayores. Tiene un papel muy importante en la actualidad, ya que permite la compatibilidad entre sistemas para compartir la información de una manera segura, fiable y fácil.

### 1.3.4 ANDROID STUDIO

Es el entorno de desarrollo integrado oficial para la plataforma *Android*. Fue anunciado el 16 de mayo de 2013 en la conferencia *Google I/O*, y reemplazó a Eclipse como su entorno de desarrollo integrado (*IDE*) oficial para el desarrollo de aplicaciones para *Android*. La primera versión estable fue publicada en diciembre de 2014.

Está basado en el software *IntelliJ IDEA* de *JetBrains* y ha sido publicado de forma gratuita a través de la Licencia *Apache 2.0*. Está disponible para las plataformas *Microsoft Windows*, *macOS* y *GNU/Linux*. Ha sido diseñado específicamente para el desarrollo de *Android*.

La razón por la que se emplea el lenguaje de programación *Java* y *XML*, se debe a que *Android* requiere ambos lenguajes para realizar cualquier desarrollo para dicho sistema operativo.

La razón principal por la que se elige desarrollar en *Android* es que no requiere ningún tipo de licenciamiento y porque su *IDE* es completamente gratuito por lo que realizar desarrollos para esta plataforma no tiene costo alguno.

### 1.3.5 ARQUITECTURA DE SERVICIOS

Los servicios proponen su propia arquitectura. Mientras que en una arquitectura monolítica la aplicación es desarrollada como una única unidad, una arquitectura de servicios funciona con un conjunto de componentes que se ejecutan de manera independiente y autónoma. Incluso cada uno de ellos puede estar en un lenguaje de programación diferente.

A continuación, se mencionan y describen brevemente algunas de las características de los servicios las cuales son:

- Características de su software: Puede ser descompuesto en diferentes partes independientes. Por ello cada uno de los servicios puede ser desplegado y modificado sin afectar a otros aspectos funcionales de la aplicación.
- Características de su organización: La manera en la que están organizados supone un contraste con el entorno monolítico. Ya que tienen en cuenta aspectos como las capacidades, necesidades y preferencias del negocio o cliente donde será implantado. En cuanto, a la arquitectura, se usan módulos multifuncionales consiguiendo la creación de un módulo común para todos ofreciendo un servicio en concreto. Sin duda, la gran ventaja es el ahorro de tiempo y la comodidad en tareas de mantenimiento evitando que, al revisar un módulo, el resto del equipo no pueda completar su jornada.
- Características de su arquitectura: Cada módulo es independiente, ya que, cada uno de ellos cuenta con su propia base de datos, es decir, no acuden todos a la misma. Así evitamos la sobrecarga y la caída de la aplicación.

Ahora que ya sabemos qué son los servicios, así como su arquitectura y características principales. A continuación, se enlistan algunas de sus ventajas y desventajas.

Ventajas:

- Equipo de trabajo mínimo
- Escalabilidad
- Funcionalidad modular, módulos independientes.
- Libertad del desarrollador para desarrollar y desplegar servicios de forma independiente.
- Uso de contenedores, permitiendo que el despliegue y el desarrollo de la aplicación sean mucho más rápido y sencillo.
- Minimizar comunicación

Desventajas:

- Alto consumo de memoria.
- Necesidad de tiempo para poder fragmentar distintos servicios.
- Complejidad de gestión de un gran número de servicios.
- Pruebas o testeos complicados al despliegue distribuido.

## 1.4 INTERACCIÓN HOMBRE MÁQUINA

Las personas en su vida cotidiana interactúan con un sinnúmero de objetos, los cuales poseen características muy evidentes a los sentidos humanos (vista, tacto, audición, olfato y gusto) para su manipulación y utilización. Uno de los objetos utilizados masivamente por los seres humanos hoy en día, son las computadoras. Las mismas, son manipuladas por un gran número de personas para todo tipo de actividades. Algunos sistemas

informáticos funcionan con poca intervención humana, pero la mayoría son interactivos, y necesitan de la interacción con los usuarios que están involucrados en la resolución de tareas.

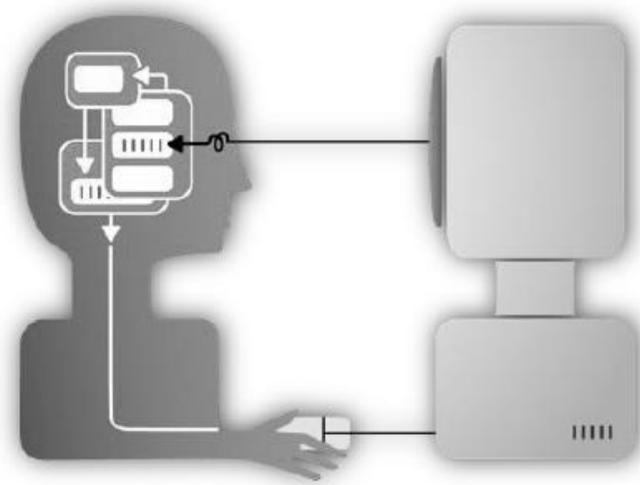


FIGURA 1.3 CONCEPTO DE IHM [1]

Hablando estrictamente, la interfaz la componen los dispositivos de entrada y salida y los programas que la hacen funcionar, pero desde un punto de vista más general, comprende todo lo que tiene que ver con la experiencia de un usuario y la computadora, como el entorno de trabajo, la organización en la que trabaja, el entrenamiento que ha tenido, la ayuda que recibe, etc.

#### 1.4.2 ESTRUCTURA DE LA INTERACCIÓN HOMBRE – MÁQUINA

La IHM está compuesta por tres elementos: la persona o usuario, la computadora y la interacción. El primero de ellos (la persona) es el o los usuarios que interactúan con el sistema, de ellas se deben conocer las capacidades físicas, habilidades perceptuales y cognitivas, edad, diferencias culturales y de personalidad, así como también las tareas que tienen que realizar y su contexto de uso. La computadora es cualquier sistema o dispositivo que cumple con determinadas características, tales como su movilidad, plataforma, capacidades de entrada y salida, o su sistema operativo. En tercer y último lugar se encuentra la interacción, que se puede definir como el intercambio que realizan entre sí la/s persona/s (o usuario/s) y la computadora.

Uno de los objetivos de este tipo de interacciones es mejorar la utilidad, eficiencia y usabilidad de la comunicación entre las personas, los sistemas y dispositivos que incluyan una computadora.

En la Figura 1.4 se muestran los distintos componentes de la IHM, y cómo interactúan entre sí. Se ve como la interacción nace del pensamiento humano, luego es transmitido mediante algún lenguaje y dispositivo definido, interpretado y procesado del lado de la computadora, para luego dar un resultado al usuario.

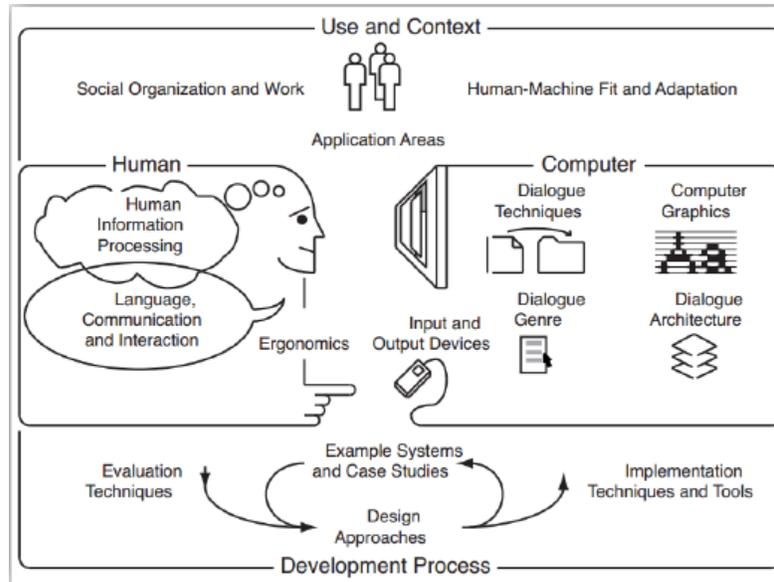


FIGURA 1.4 COMPONENTES DE UNA IHM [1]

### 1.4.3 OBJETIVOS DE LA INTERACCIÓN HUMANO-COMPUTADORA

Mc Craken considera que la IHM tiene como propósito hacer que la experiencia de las personas que usan la computadora sea productiva, eficiente en tiempo de uso y placentera [2]. Por su parte, Eason propone establecer que: “El objetivo de la IHM es desarrollar o mejorar la seguridad, utilidad, eficiencia, usabilidad y apariencia de los sistemas que incluyan a la computadora” [36].

- Seguridad, puede ser “seguridad del usuario”, “seguridad de los datos” o ambos.
- Utilidad, se refiere a los servicios que la aplicación provee a su usuario, se dice que una aplicación es útil si a través de ella dicho usuario puede realizar sus actividades.
- Eficacia, existe eficacia si el usuario puede cumplir con su objetivo o finaliza la operación deseada.

La utilidad y la eficacia son distintos aspectos de un sistema interactivo.

Una aplicación puede proveer toda la información, instrucciones y soporte para realizar una compra (utilidad), sin embargo, si los usuarios no pueden visualizar como encontrar los artículos que se desean comprar, entonces la aplicación carece de eficacia.

- Eficiencia, es una medida de que tan rápido los usuarios pueden cumplir sus objetivos o finalizar su trabajo cuando usan un sistema interactivo.
- Usabilidad, es una descripción de la facilidad de aprendizaje y facilidad de uso de la aplicación por parte del usuario.
- Apariencia, es una descripción de que tanto le agrada el sistema al usuario, incluyendo consideraciones tales como la primera impresión y la satisfacción a largo plazo.

En la Figura 1.5 se muestran los objetivos de la IHM; como puede notarse, se debe realizar una distribución del esfuerzo para el desarrollo de un proyecto de acuerdo con la naturaleza del problema.



FIGURA 1.5 OBJETIVOS DE IHM [2]

Resulta un tanto complejo establecer alguna prioridad para la distribución del esfuerzo en el desarrollo de aplicaciones para la elaboración del proyecto que se está planteando.

#### 1.4.4 DISEÑO

Es la actividad está destinada a conseguir la creación de objetos útiles. Actualmente se pretende actuar sobre el entorno físico del hombre para mejorarlo en su conjunto

Algunos principios definidos y fundamentales para el diseño de las interfaces son:

- Simplicidad. El programa debe ser de manejo fácil (buena usabilidad). Esto no debe implicar la pérdida de funcionalidad.
- Apoyo. El control del programa recae sobre el usuario. El sistema debe ofrecerle ayuda para facilitarle la labor.
- Familiaridad. Se debe diseñar la interfaz teniendo en cuenta los conocimientos previos del usuario.
- Evidencia. Se deben diseñar los controles y otros objetos visibles e intuitivos, por ejemplo, usando representaciones del mundo real.
- Estímulo. Las acciones de los usuarios deben producir los resultados que ellos esperan, es decir, ser previsible. También se deberían poder deshacer, ser reversibles.
- Satisfacción. El usuario debería sentir que está progresando y haciendo avances, debe haber retroalimentación de parte de la interfaz.
- Disponibilidad. Hacer que todos los objetos estén disponibles para que el usuario los use en cualquier momento.
- Seguridad en el uso. Se refiere a evitar errores del usuario. Para ello la interfaz puede ofrecerle ayuda de forma automática o si el usuario la solicita.
- Versatilidad. El sistema debe soportar diferentes formas de interacción para que el usuario pueda elegir el método más adecuado a su situación.
- Personalización. La interfaz debe ser personalizable para que el usuario la adapte según sus necesidades.

- Afinidad. Permitir que a través de un buen diseño visual los objetos sean afines a otros de la realidad cotidiana, esto contribuye a la familiarización del usuario con la interfaz.

## 1.5 METODOLOGÍA DE DESARROLLO CENTRADA EN EL USUARIO

Por la naturaleza del problema de estudio, se utiliza la metodología de desarrollo basada en el usuario, la cual está basada en los principios de IHM. Dicho enfoque de la IHM está orientado a la creación de sistemas de cómputo interactivo (software multimedia, navegadores de internet, software colaborativo, etc.). Las consideraciones mayores con los sistemas de cómputo interactivo incluyen facilidades para usuario con limitaciones físicas y consideraciones del medio ambiente donde será ubicado el sistema. El analista IHM escucha a los usuarios y comunica sus necesidades a los responsables del desarrollo para su implementación [2].

La metodología de desarrollo centrada en el usuario es esencial para la creación de las interfaces de usuario [2]. Una interfaz de usuario conjunta esos aspectos del sistema con los que el usuario tiene contacto [13].

La metodología de desarrollo centrada en el usuario es útil para la creación de cualquier tipo de interfaz, desde programas de oficina, hasta videojuego y sitios WEB [2]. Esta metodología de desarrollo difiere de las metodologías de la ingeniería de software en tres aspectos específicos [2]:

1. El desarrollo es centrado en el usuario y no en los datos, el desarrollo centrado en el usuario involucra al usuario en el proceso de implementación tanto como sea posible, con el objetivo de crear una interfaz que cubra las expectativas del usuario. Para ello se realizan diversas tareas tales como: observar al usuario mientras trabaja, incluir al usuario en el equipo de diseño y solicitar al usuario realizar pruebas con el sistema para obtener sus observaciones y aplicar los cambios necesarios.
2. El desarrollo centrado en el usuario es interdisciplinario y se base en el conocimiento de diversas áreas como el diseño gráfico, la psicología cognitiva, la ergonomía y las ciencias computacionales entre otras. La Figura 1.6 ilustra esto con mayor detalle.
3. La metodología es iterativa e involucra tantas pruebas y revisiones como sea posible.

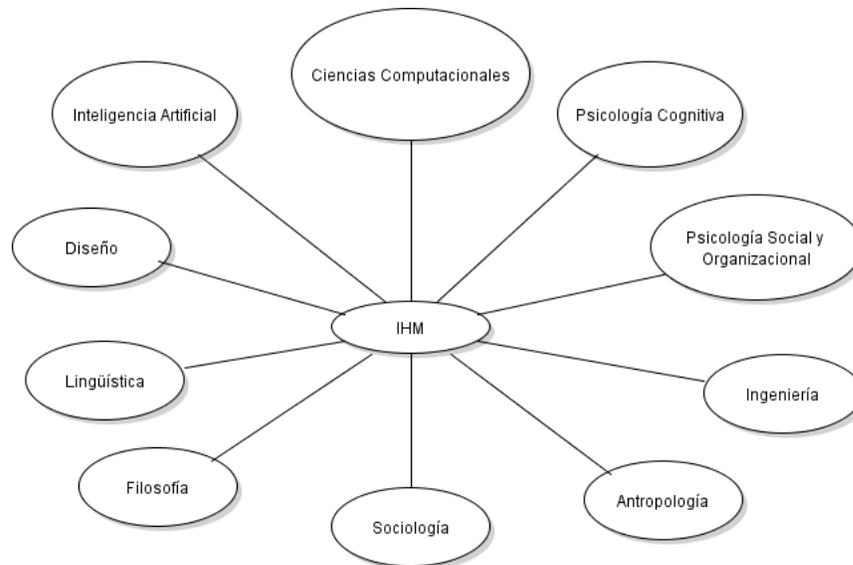


FIGURA 1.6 DISCIPLINAS INVOLUCRADAS EN LA IHM [2]

La metodología de desarrollo centrada en el usuario consta de las siguientes etapas [2]:

1. Análisis
  - a. Análisis de necesidades
  - b. Análisis del usuario y sus tareas
  - c. Análisis funcional
  - d. Análisis de requerimientos
2. Especificaciones de usabilidad/Métricas de usabilidad
3. Diseño
4. Prototipo
5. Evaluación

A continuación, se dará una breve descripción de cada etapa de la metodología de desarrollo centrada en el usuario.

- a) Análisis de necesidades. Resume la naturaleza y el propósito del sistema interactivo que se desea desarrollar, También describe el tipo de sistema y menciona a la gente que servirá, así como los beneficios que brindará.
- b) Análisis del usuario y sus tareas. Se identifica a la gente que usará la aplicación, incluyendo una descripción general como: edad, nivel de educación y experiencia con las computadoras. El análisis de las tareas consiste en el observar el tipo de trabajo que el usuario realizará con la aplicación, así las tareas o actividades que ellos llevan a cabo para realizar su trabajo. Varias aplicaciones fallan porque el equipo de trabajo no dedico tiempo para aclarar quienes son sus usuarios o que es que ellos desean hacer [13].
- c) Análisis funcional. Se identifica la funcionalidad y los servicios de cómputo que los usuarios necesitan para poder realizar sus tareas. Usualmente hay cierta relación entra las funciones a realizar y las tareas que va a realizar el usuario. En esta etapa es donde se decide que aspecto de las tareas deben ser automatizados y cuáles deben ser realizados por el usuario.
- d) Análisis de requerimientos. Describe las especificaciones requeridas para implementar un sistema. Dependiendo de la aplicación, la especificación formal puede incluir diccionario de datos, diagramas de entidad relación o modelado orientado a objetos. El análisis de requerimientos es del mismo tipo de especificación que se desarrolla en el análisis de sistema clásico o en la ingeniería de software, para más información se puede consultar [12].
- e) Especificaciones de usabilidad / Métricas de usabilidad. Las especificaciones de usabilidad son indicadores que ayudan a determinar la efectividad de la aplicación interactiva. Las especificaciones de usabilidad incluyen *medidas de desempeño*, tales como numero de tareas terminadas y número de errores, las cuales son directamente observables de la conducta del usuario. Las especificaciones usabilidad también incluyen *medidas de preferencia*, las cuales proporcionan indicaciones acerca de la opinión del usuario respecto a la aplicación. Por ejemplo “la primera impresión” y “satisfacción total”.
- f) Diseño. En esta etapa se decide sobre la organización y la apariencia de la aplicación. Cuando se realiza el diseño, se identifica el contenido de la interfaz y se organiza acorde a las expectativas del usuario. La organización del contenido es uno de los elementos más críticos para el éxito de la aplicación. El diseño también incluye la distribución de pantallas individuales, así como usar las

técnicas de organización visual para crear claridad y consistencia entre ellas. A su vez en el diseño se determinan las funciones de navegación de la interfaz. En esta etapa se inicia la creación de bosquejos sobre las interfaces de la aplicación, una vez que se sabe quién es el usuario y que es lo que se desea realizar con la aplicación.

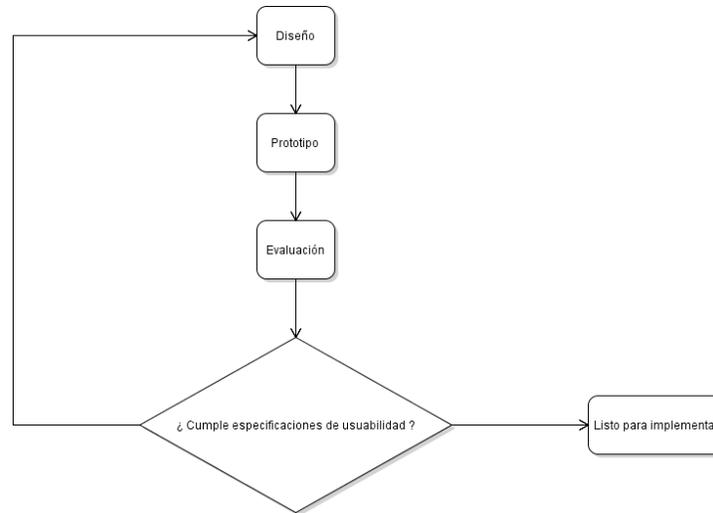
**TABLA 1.3 CLASIFICACIÓN DE PROTOTIPOS**

a) Por su representación	Prototipo global Es un modelo completo de la aplicación.
	Prototipo local Es un modelo de una sola parte de la aplicación.
b) Por su utilidad	Prototipo evolutivo Es un modelo que será refinado hasta llegar a la aplicación final.
	Prototipo borrador Es un modelo de implementación, que es deshecho una vez que la aplicación sea completada.
c) Por su calidad	Prototipo de alta fidelidad Es un modelo que muestra la apariencia y la funcionalidad de la aplicación final.
	Prototipo de baja fidelidad Es un modelo que tiene una mínima aproximación a la aplicación final, basado en papel.

- g) Prototipo. El prototipo es un modelo de referencia o patrón, en base al cual será implementada la aplicación. Los tipos de prototipos se clasifica de diversas formas, como se puede apreciar en la Tabla 1.3.
- h) Evaluación. En esta etapa se realiza la prueba del prototipo, con el fin de determinar si se han cumplido las especificaciones de usabilidad establecidas en la etapa de análisis. Se pueden realizar dos tipos de evaluaciones:
- a. Evaluación basada en usuario, en la cual se solicita a los usuarios realiza tareas representativas utilizando el prototipo. Al mismo tiempo que el usuario realiza cada tarea, un observador toma nota de problemas que ocurran.
  - b. Evaluación basada en expertos, en la cual un grupo de expertos en usabilidad critican el prototipo

Ambas formas de evaluación tienen beneficios, sin embargo, para fines de la IHM se sugiere la evaluación basada en el usuario, Una vez obtenido los resultados de la evaluación se vuelve a rediseñar las secciones del prototipo y se realizan nuevamente las pruebas.

El desarrollo centrado en el usuario es iterativo por naturaleza tiene un ciclo de repetición en las etapas de diseño y evaluación, como se puede apreciar en la Figura 1.7.



**FIGURA 1.7 CICLO ITERATIVO DE LA METODOLOGÍA DE DESARROLLO CENTRADA EN EL USUARIO [2]**

Idealmente se continua el ciclo diseño-prototipo-evaluación hasta que se hayan cumplido las especificaciones de usabilidad.

- i) Consideraciones adicionales. Con el fin de mejorar el sistema interactivo, la metodología de desarrollo centrada en el usuario se recomienda considerar la aplicación efectiva del color, tipografía y multimedia. De igual forma, examinar los efectos que las discapacidades de los usuarios pueden tener para interactuar con la aplicación y realizar sugerencias en el diseño de la aplicación para entender a los usuarios con necesidades especiales.

## 1.6 NATURALEZA DE LAS INTERFACES GESTUALES

### 1.6.1 GESTO

Gestos y reconocimiento de gestos son términos que se usan a menudo en las discusiones sobre la interacción humano-computadora. Para la mayoría de las personas, el término incorpora reconocimiento de caracteres, reconocimiento de los correctores de símbolos, y varios tipos de interacción. De hecho, toda acción física involucra un gesto para que esta sea llevada a cabo. Asimismo, la naturaleza de ese gesto es un importante componente que establece la sensación que le queremos dar a la acción.

Antes de dar una definición formal de qué es un gesto, se debe hablar de la comunicación no verbal, que es, básicamente, comunicar sin palabras a través de múltiples canales de comunicación. Las expresiones no verbales son las acciones diferentes al habla que se realizan durante una comunicación, esto se puede interpretar como una aproximación a la definición de gesto.

Si involucramos en la definición de gesto a un sistema interactivo, podemos decir que un gesto es cualquier movimiento físico que un sistema interactivo detecta y el cual genera una respuesta de este, como se muestra en la Figura 1.8.



FIGURA 1.8 EJEMPLO DE UN GESTO

Por supuesto que, si analizamos esta definición, no es difícil de darse cuenta de que para el uso de gestos se requiera de dispositivos de entrada. Los gestos involucran un gran número de grados de libertad y restricciones para con el usuario.

Los grados de libertad en el cuerpo humano, están determinados por los ejes de movimiento que posee una articulación y a su vez estos estarán condicionados por la forma anatómica de dicha articulación. Al realizar un mismo movimiento que involucra una articulación determinada, por ejemplo, el hombro, existen ejercicios que darán un grado de libertad porque utiliza solo uno de los ejes de movimiento del hombro y los demás son anulados, esto sucedería, por ejemplo, con una máquina.

Seis grados de libertad (6DOF) se refiere al movimiento en un espacio tridimensional, es decir, la capacidad de moverse hacia delante/atrás, arriba/abajo, izquierda/derecha (traslación en tres ejes perpendiculares), combinados con la rotación sobre tres ejes perpendiculares. El movimiento a lo largo de cada uno de los ejes es independiente de los otros, y cada uno es independiente de la rotación sobre cualquiera de los ejes, el movimiento de hecho tiene seis grados de libertad.

Tres grados de libertad (3DOF) significa solo el movimiento en tres dimensiones, pero no incluye la rotación.

### 1.6.2 TIPOS DE GESTOS

Existen diversos tipos de gestos, sin embargo, estos se han clasificado en cinco tipos que son: gestos emblemáticos, gestos ilustrativos, gestos pantógrafos, gestos reguladores de interacción y gestos de adaptación.

A continuación, se describen los gestos anteriormente mencionados.

- A. Gestos emblemáticos o emblemas, son señales emitidas intencionalmente. Su significado es específico y muy claro, ya que el gesto representa una palabra o conjunto de palabras bien conocidas. Por lo tanto, son gestos traducibles directamente en palabras.
- B. Gestos ilustrativos o ilustradores, se producen durante la comunicación verbal. Sirven para ilustrar lo que se está diciendo. Son gestos conscientes que varían en gran medida en función de la cultura. Son gestos unidos al lenguaje, pero a diferencia de los emblemas, no tiene un significado directamente traducible, la palabra a la que van unidos no les da su significado. Este tipo de gesto sirve a esa palabra no la significa. La forma de servirla reside en su capacidad para recalcar lo que se dice, enfatizar o imponer un ritmo a la palabra que esta por sí misma no tendría.

- C. Gestos pantógrafos, este tipo de gesto cumple un papel similar a los ilustradores y por ello se pueden confundir. Es similar en el sentido en que también acompañan a la palabra, y le confieren un mayor dinamismo. Pero difieren en que este tipo de gestos reflejan el estado emotivo de la persona, mientras que el ilustrador es emocionalmente neutro.
- D. Gestos reguladores de la interacción: Son movimientos producidos por quién habla o por quién escucha, con la finalidad de regular las intervenciones en la interacción. Son signos para tomar el relevo en la conversación, que tienen también un importante papel al inicio o finalización de la interacción.
- E. Los gestos adaptadores son aquellos movimientos que realizamos tocando nuestro propio cuerpo o manipulando algún objeto, como ajustarnos los puños de la camisa o llevarnos la mano a la boca. Están vinculados fundamentalmente al manejo de las emociones

### 1.6.3 CLASIFICACIÓN POR DISPOSITIVOS DE ENTRADA

La capacidad de realizar un seguimiento de los movimientos de una persona y determinar qué gestos pueden ser realizados a través de diversas herramientas. Aunque hay una gran cantidad de investigaciones realizadas en la imagen y/o vídeo basado en el reconocimiento de gestos, hay una variación dentro de las herramientas y entornos utilizados entre implementaciones.

Guantes cableados. Estos pueden proporcionar datos de entrada a la computadora sobre la posición y la rotación de las manos utilizando dispositivos de seguimiento magnético o inercial. Además, algunos guantes pueden detectar los dedos con un alto grado de precisión (5-10 grados), o incluso proporcionar retroalimentación háptica del usuario, lo cual se convierte en una simulación del sentido del tacto.

El primer dispositivo de seguimiento de mano de tipo guante disponible en el mercado fue el *Data Glove*, es un dispositivo de tipo guante que puede detectar la posición de la mano, el movimiento y los dedos, por separado.

Esto utiliza cables de fibra óptica que se colocan en la parte posterior de la mano. Pulsos de luz son creados y cuando se doblan los dedos, fugas de luz a través de pequeñas grietas y las pérdidas que se registran, dan una aproximación de la actitud de la mano.

Cámaras de profundidad consciente en el uso de cámaras especializadas, como la luz estructurada o cámaras de tiempo de vuelo, se puede generar un mapa de profundidad de lo que se ve a través de la cámara a una distancia corta, y utilizar estos datos para aproximarse a una representación en 3D de lo que se está viendo. Estos pueden ser eficaces para la detección de gestos con las manos, debido a sus capacidades de corto alcance.

Cámara estereoscópica, nombrada así debido a la visión estereoscópica humana (3D), es una cámara capaz de capturar fotografías en tres dimensiones. La visión binocular humana, produce dos imágenes (una para cada ojo) que luego se mezclan en el cerebro creando la imagen 3D. Estas cámaras, intentan imitar este comportamiento, utilizando dos cámaras separadas estratégicamente, captando la fotografía en el mismo instante, y como resultado se obtienen las imágenes 3D.

Gestos basados en controlador. Estos controladores actúan como una extensión del cuerpo, de modo que, cuando se llevan a cabo gestos, algunos de sus movimientos pueden ser capturados mediante *software*. Los gestos del ratón son un ejemplo, donde el movimiento del ratón se correlaciona con un símbolo que está dibujado por la mano de una persona, al igual que el mando de Wii.

Una cámara 2D estándar se puede utilizar para el reconocimiento de gestos donde los recursos y/o el medio ambiente no serían convenientes para otras formas de reconocimiento basada en imágenes. Anteriormente se pensaba que una sola cámara puede no ser tan eficaz como las cámaras de profundidad, pero algunas compañías están desafiando esta teoría. La tecnología de reconocimiento de gestos basada en software con una cámara 2D estándar puede detectar gestos robustos de mano, señales con las manos, así como las manos o los dedos con una gran precisión.

Dependiendo del tipo de datos de entrada, el enfoque para interpretar un gesto podría hacerse de diferentes maneras. Sin embargo, la mayoría de las técnicas se basan en indicadores clave representados en un sistema de coordenadas 3D, basado en el movimiento relativo de estos, el gesto puede ser detectado con una alta precisión, dependiendo de la calidad de la entrada y el enfoque del algoritmo.

#### 1.6.4 CONSIDERACIONES DE DISEÑO DE UNA INTERFAZ GESTUAL

Al momento de diseñar la interfaz de usuario diferentes aspectos deben ser tomados en consideración especialmente porque diseñar una interfaz basada en gestos sin contacto es diferente a diseñar una *GUI* y simplemente incorporarle gestos [14]. ¿Qué gestos utilizar?, ¿Qué acciones se llevarán a cabo con qué gestos?, ¿Cuál es el espacio en el que interactuarán los usuarios?, etc. En las siguientes subsecciones se intentará responder estas preguntas.

Un buen punto de partida para diseñar una interfaz es tener en cuenta los principios de diseño de interfaces utilizados en IHM. Estos son abstracciones que ayudan a los diseñadores a pensar en los aspectos de sus diseños [3] y que pueden conducir a tener una aplicación con mayor usabilidad. A continuación, se describen algunos de ellos tratando de orientarlos a la interacción sin contacto.

**Asequibilidad.** Es el término usado para referirse al atributo de un objeto que permite a los usuarios conocer cómo utilizarlo [3]

**Visibilidad.** Este principio señala que las funciones que se espera sean más utilizadas por el usuario estén claramente visibles, indicando qué se puede hacer [3]. De esta manera, la forma de controlar una aplicación queda notoriamente visible; además, en las *GUIs* el uso de menús es muy importante para que más acciones sean visibles.

**Mapping.** Se refiere a la relación existente entre los controles y sus efectos [3]. Por ejemplo, en un software para reproducir vídeos, un botón con una imagen de un triángulo apuntando hacia la derecha puede tener el efecto de reproducir un vídeo.

**Retroalimentación.** Se entiende como retroalimentación a la información que se envía de vuelta al usuario para informarle sobre la acción que está realizando y/o se ha completado [3]. Como parte de una interfaz sin contacto, el usuario puede recibir retroalimentación visual, auditivo, afectivo y háptico (o táctil). La retroalimentación afectiva se refiere a dar al usuario alguna información de carácter emocional y puede ser entregado de varias maneras tales como expresiones faciales.

Un gran desafío al momento de diseñar una interfaz basada en gestos es el conjunto de gestos que se utilizarán. Este conjunto de gestos es conocido con el nombre de vocabulario de gestos (traducido del inglés *gesture vocabulary* [15]). Los gestos seleccionados deberían ser fáciles de realizar para el usuario y la aplicación debe poder reconocerlos.

Por otro lado, es necesario tener en cuenta que la cantidad de gestos a utilizar no puede ser ilimitada. Usar una mayor cantidad de gestos no sólo puede representar un problema para el usuario porque tiene que recordarlos sino también para que la aplicación pueda reconocerlos [16]. Varios estudios relacionados a la

cantidad de gestos que el usuario puede recordar han sido realizados [17]; sin embargo, no existe un número fijo. Por otro lado, teniendo en cuenta la capacidad de la memoria de trabajo un número razonable podría ser  $7 \pm 2$  gestos [18].

Una forma de superar los límites de la cantidad de gestos que los usuarios pueden recordar es utilizar gestos definidos por ellos mismos [19]. El problema de esta opción es que no es adecuada para aplicaciones enfocadas a las salas de clases porque varias personas utilizarán la misma aplicación, incluso al mismo tiempo, y los gestos definidos por uno pueden no ser los deseados y/o recordados por otros.

Sin lugar a duda, la forma más rápida de seleccionar los gestos es basarse en trabajos similares. En la Tabla 1.4 se muestran algunos de los gestos utilizados, así como también las acciones para las cuales fueron utilizados.

**TABLA 1.4 GESTOS USUALMENTE EMPLEADOS EN OTRAS INTERFACES**

<b>Gestos</b>	<b>Descripción</b>	<b>Acciones/ usos</b>
<b><i>Draw</i></b>	Dibujar en el aire figuras, letras, números.	Ejecutar comandos asociados al dibujo o ingresar algún dato [21]; autenticación de usuarios [22].
<b><i>Grip</i></b>	Cerrar la mano (hacer puño)	“Agarrar” un objeto para desplazarlo [23]; seleccionar botones/opciones.
<b><i>Hold o hover</i></b>	Mantener la mano en una posición.	Seleccionar botones/opciones [23;26-27].
<b><i>Pull</i></b>	Mover la mano del frente hacia el cuerpo.	Seleccionar botones/opciones [24].
<b><i>Release</i></b>	Abrir la mano.	“Soltar” el objeto previamente “agarrado” [23].
<b><i>Swipe (horizontal y vertical)</i></b>	Mover la mano de derecha a izquierda y viceversa (horizontal), o de arriba hacia abajo y viceversa (vertical).	Navegar entre opciones [23-24].
<b><i>Tap o Push</i></b>	Empujar o mover la mano hacia el frente.	Seleccionar botones/opciones [23].
<b><i>Wave</i></b>	Señal de “hola”.	Identificar usuarios [25]. Iniciar o reiniciar [26]. Seleccionar botones/opciones [26].

Los estudios incluidos en la Tabla 1.4 estuvieron enfocados a desarrollar algún tipo de aplicación para algún propósito particular y no estudiar los gestos usados (por ejemplo, con base en el tiempo requerido para realizarlos o su dificultad). Además, los gestos mencionados son realizados con una mano. Es importante mencionar también que la mayoría de estos gestos han sido utilizados en sistemas comerciales como puede verse en [20].

Para realizar el análisis y diseño de interfaces gestuales es muy importante saber qué se quiere realizar, para qué va a ser destinada esa interfaz y cómo se introducirá al contexto elegido previamente.

En el capítulo II, se realiza la parte del análisis para poder realizar el desarrollo del primer prototipo de la interfaz gestual.

Adicionalmente, también se incluye desarrollo del prototipo de la interfaz gestual, así como también las tecnologías que se emplearon para su programación.

## CAPÍTULO II. DESARROLLO DE LA APLICACIÓN MÓVIL

---

*En este capítulo se realiza el análisis de la funcionalidad que tendrá la interfaz gestual y también incluirá el desarrollo de esta.*

### 2.1 ANÁLISIS

En esta sección, se desarrolla la etapa de análisis de la metodología de desarrollo centrada en el usuario para la creación de un prototipo de una aplicación móvil con interfaz gestual, la cual ayude a los usuarios a realizar el envío de una alarma cuando se encuentren en una situación de riesgo.

#### **Análisis de necesidades**

Lamentablemente, uno de los problemas actuales a los que se enfrentan los transeúntes en la vía pública, es el estar expuestos a ser asaltados. Sin embargo, más grave aún es que estas personas no cuenten con un medio para realizar una notificación a algún conocido o familiar, quien pueda ayudarles en dichos momentos de riesgo.

Más aún, inclusive en situaciones de menor riesgo, una herramienta que permita enviar notificaciones para recibir ayuda ante un incidente puede ser de gran ayuda.

#### **Análisis del usuario y sus tareas**

La aplicación pretendida se orienta a usuarios transeúntes, quienes hacen uso de la vía pública para realizar actividades cotidianas como: dirigirse al trabajo, ir a la escuela, ir de compras, actividades deportivas y recreativas. Quienes además son personas con capacidad de actuar de manera autónoma al realizar dichas actividades y que generalmente utilizan un teléfono celular.

#### **Análisis funcional**

Lo que se busca realizar en este desarrollo es crear una aplicación *Android* que puede realizar el envío de alarmas por medio de una interfaz gestual cuando un usuario se enfrente una situación de riesgo como un asalto en la vía pública.

La razón para que la aplicación móvil tenga una interfaz gestual habilitada, es para que el usuario no recurra a una interfaz gráfica de usuario convencional, la cual es una interfaz inadecuada para usuarios que enfrentan situaciones de riesgo, ya que esta requiere la interacción con elementos gráficos de la pantalla del teléfono celular.

Además, durante un asalto la persona agraviada enfrenta niveles altos de estrés y nerviosismo por dicha situación, lo cual dificulta que el usuario pueda realizar tareas complejas con el teléfono celular. Es por esta razón, se considera que una interfaz de usuario no estándar es más adecuada, porque con este tipo de interfaz solo necesita realizar un movimiento para enviar la alarma.

### Análisis de requerimientos

Las tecnologías usadas para el desarrollo de la aplicación móvil con interfaz gestual son:

- *Android* (Versión 5 como mínimo)
- *SQL Server 2012*
- GPS
- *SMS*
- Acelerómetro

Para el correcto funcionamiento de la aplicación móvil, requiere que el dispositivo móvil a usar cuente con la versión 5 de *Android* como mínimo, ya que esta tiene cambios que mejoran el desempeño de la aplicación. A continuación, se describen algunos de los cambios más significativos:

- *Material Design*, Este aspecto incluye un diseño intrépido, colorido, y sensible para la interfaz de usuario la cual genera experiencias coherentes e intuitivas en todos los dispositivos. Un movimiento de respuesta más natural, iluminación y sombras realistas, así como también elementos visuales familiares que hacen que sea más fácil de navegar en el dispositivo. Nuevos colores vivos, tipografía e imágenes de ayuda de borde a borde de enfocar su atención.
- Notificaciones, Las nuevas formas de controlar cuándo y cómo se reciben mensajes, ya que incluye la capacidad de ocultar contenido sensible para estas notificaciones. Se puede programar el tiempo durante el cual sólo las notificaciones de prioridad aparecen.
- Batería, La característica de ahorro de batería extiende el uso de dispositivos de hasta 90 minutos. El tiempo estimado de batería restante aparece cuando el dispositivo está enchufado.

La razón la cual se eligen los puntos anteriores para el desarrollo son:

- Porque la interfaz gestual se ejecuta en segundo plano, lo cual consume batería adicional.
- Al mejorar los elementos visuales, esto permite usar elementos llamativos o agradables para el usuario, adicionalmente se pueden diseñar interfaces gráficas más amigables e intuitivas para el usuario.
- En cuanto al manejo de las notificaciones, permite personalizar los mensajes sobre todo cuando se realiza el envío de la alarma.

Otras características que deben tomar cuenta para el dispositivo son:

- El dispositivo debe contar con sensor de GPS, ya que sin este no se podrían obtener las coordenadas, de donde se está enviando la alarma.
- El dispositivo debe contar con sensor del acelerómetro debido a que este sensor el dispositivo de entrada a la interfaz gestual.
- El dispositivo debe estar conectado a una red que tenga salida a internet o tener datos móviles o en su defecto estar conectado a una VPN para poder hacer el envío de la alarma.

## 2.2 MÉTRICAS DE USABILIDAD

Las métricas de usabilidad son medidas precisas para evaluar la calidad de un sistema, estas pueden ser tanto cualitativas como cuantitativas, lo que se busca medir con las métricas de usabilidad es la eficacia, la eficiencia y la satisfacción. Algunas de las métricas sugeridas por la norma ISO/IEC 9126-4 para medir estos puntos se muestran en la Tabla 2.1

**TABLA 2.1 MÉTRICAS SUGERIDAS**

Objetivo	Métricas
Eficacia	<ul style="list-style-type: none"> <li>• Cantidad de tareas realizadas.</li> <li>• Porcentaje de funciones relevantes utilizadas.</li> <li>• Porcentaje de tareas completadas con éxito en el primer intento.</li> <li>• Cantidad de errores persistentes o por repetición.</li> <li>• Cantidad de errores por unidad de tiempo.</li> <li>• Porcentaje de usuarios capaces de completar con éxito la tarea.</li> <li>• Cantidad de solicitudes de asistencia (hacia el moderador).</li> <li>• Medida objetiva de la calidad de la producción.</li> <li>• Medida objetiva de la cantidad de producción.</li> <li>• Porcentaje de usuarios que pueden llevar a cabo tareas clave sin leer un manual.</li> </ul>
Eficiencia	<ul style="list-style-type: none"> <li>• Tiempo para ejecutar un conjunto particular de instrucciones.</li> <li>• Tiempo empleado en el primer intento.</li> <li>• Tiempo para realizar una determinada tarea</li> <li>• Tiempo para realizar la tarea en comparación con un experto.</li> <li>• Tiempo para alcanzar el rendimiento de expertos.</li> <li>• Número de pulsaciones de teclas adoptadas para lograr la tarea.</li> <li>• El tiempo dedicado a la corrección de errores.</li> <li>• Número de iconos recordados después de la finalización de la tarea.</li> <li>• Porcentaje de tiempo dedicado al uso del manual.</li> <li>• El tiempo dedicado en volver a aprender las funciones.</li> </ul>
Satisfacción	<ul style="list-style-type: none"> <li>• Porcentaje de los clientes que califican el producto como satisfactorio o placentero.</li> <li>• Cambio de uso voluntario.</li> <li>• Porcentaje de usuarios que se sienten “en control” del producto.</li> <li>• Porcentaje de clientes que se lo recomendaría a un amigo después de su uso en un corto periodo de tiempo.</li> <li>• Porcentaje de clientes que declaran accesible el producto.</li> </ul>

Para evaluación de la aplicación móvil desarrollada en esta tesis, se consideran tanto métricas de usabilidad de rendimiento como de preferencia.

En la Tabla 2.2 se muestran las métricas de usabilidad que se emplean para medir la usabilidad de la aplicación móvil.

TABLA 2.2 MÉTRICAS DE USABILIDAD A EMPLEAR EN LA PRUEBA DE USABILIDAD

Métricas de usabilidad	
Del usuario (instalación/configuración de la aplicación y envío de la alarma)	Número de tareas completadas exitosamente.
	Facilidad de uso de la aplicación por medio de una escala de preferencias.
	Tiempo que toma realizar la tarea asignada. <ul style="list-style-type: none"> <li>• <b>Registro de alarma:</b> diferencia entre la hora de envío desde el celular y la hora de registro en la <i>DB</i>.</li> <li>• <b>Envío de notificación:</b> diferencia entre la hora de registro en la base de datos y la hora de recepción del mensaje de alarma en el celular del contacto.</li> <li>• <b>Instalación:</b> diferencia entre la hora de inicio y la hora de término de la instalación de la aplicación.</li> <li>• <b>Configuración:</b> Diferencia entre la hora de inicio y la hora de conclusión de configuración de la aplicación.</li> </ul>

Se describen los motivos por las cuales se escogen las métricas definidas en la Tabla 2.2:

- a) Del usuario (instalación/configuración de la aplicación y envío de la alarma)
  - a. Número de tareas completadas exitosamente, se elige esta métrica para medir el número de notificaciones enviadas por el usuario y recibidas por el contacto preferente elegido.
  - b. Facilidad de uso de la aplicación, se considera esta métrica para medir el grado satisfacción que tuvieron los usuarios al estar utilizando la aplicación gestual.
  - c. Tiempo que toma realizar las tareas asignadas, con esta métrica se busca medir los tiempos que se desglosan en la Tabla 2.2, para buscar mejorar los procesos los cuales están involucrados.

### 2.3 DISEÑO

Para el desarrollo de la aplicación se plantean tres tareas principales las cuales son: instalación, configuración y notificación. Cada una de estas tareas cuenta con diversas actividades que son necesarias para completar esta dicha tarea, en la Figura 2.1 se observa cada tarea con sus correspondientes actividades.

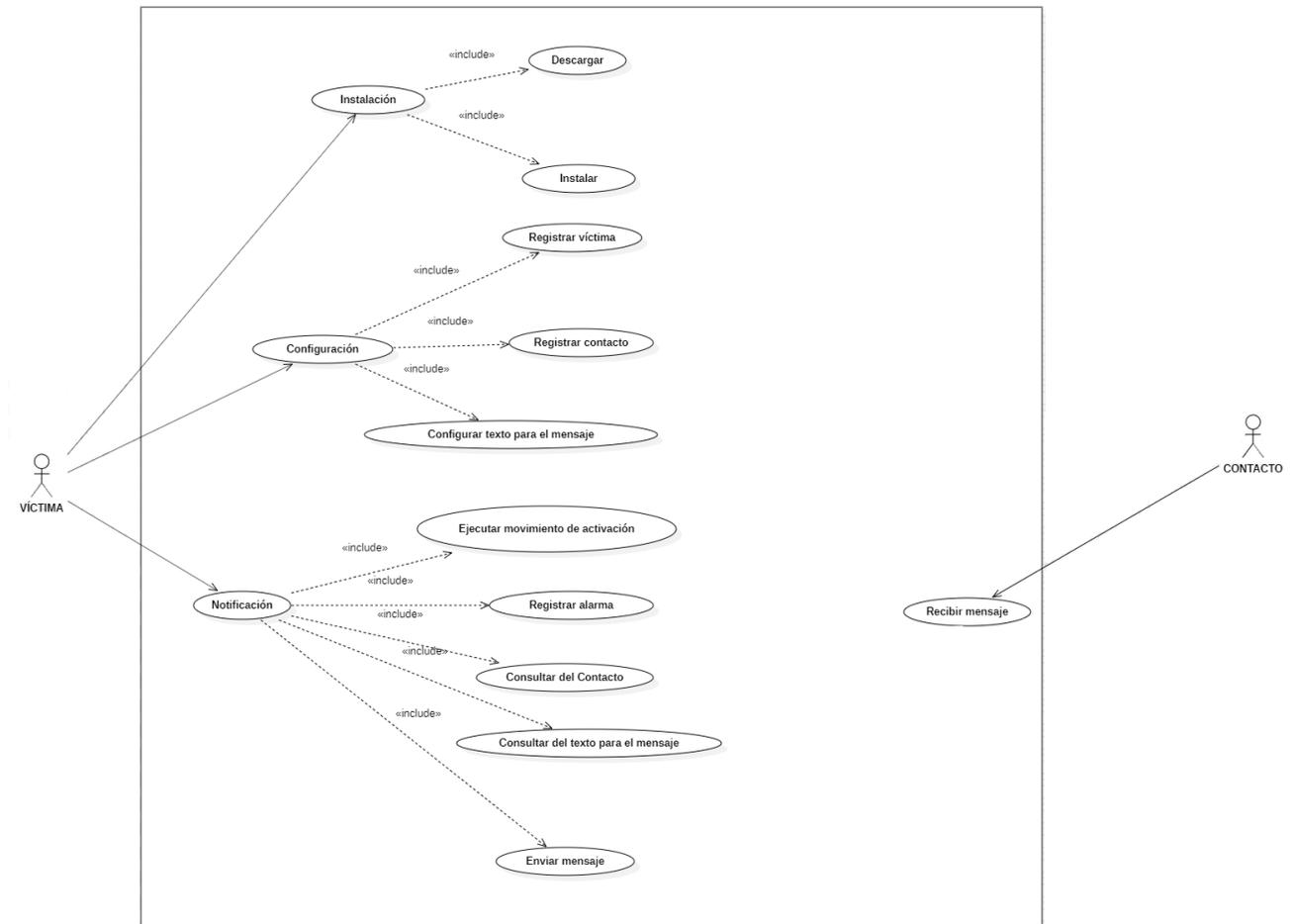


FIGURA 2.1 CASOS DE USO PRINCIPALES DE LA APLICACIÓN GESTUAL

Para la tarea de instalación se consideran las actividades de descarga e instalación, porque ambas actividades son necesarias para completar la instalación de la aplicación.

Para la actividad de descarga de la aplicación se considera a *Google Drive* como servidor de descarga, ya que, de momento, la aplicación aún no cuenta con la autorización correspondiente para ser subida a la *Play Store*. La aplicación se encuentra en un apartado público, así se podrá ingresar en cualquier momento a *Google Drive* para descargarla.

Para la tarea de configuración se consideran las actividades de registro del usuario o víctima, registro de un contacto de emergencia y la configuración de un mensaje personalizado que es enviado cuando del usuario se enfrenta a una situación de riesgo. Adicionalmente se agregan las siguientes funcionalidades a la configuración:

1. Registro e Ingreso
2. Alta de contactos
3. Consulta de contactos
4. Edición de contactos
5. Configuración personalizada

Para el registro e ingreso, se piensa en dos eventos principales: el primero es cuando el usuario ingresa por primera vez a la aplicación y el segundo evento es cuando el usuario ya se ha registrado en la aplicación.

Para el primer caso, cuando el usuario ingresa por primera vez este debe ingresar sus datos personales tales como: nombre completo, edad, género, correo electrónico. Adicionalmente se solicita que ingrese los datos de un contacto de emergencia para realizar la notificación en caso de que el usuario se enfrente a un asalto en la vía pública, los datos solicitados del contacto son: nombre completo, correo electrónico y número telefónico.

El alta de contactos se usa para registrar contactos de emergencia, con la finalidad de tener a los contactos en la aplicación y posteriormente se pueda elegir a uno de estos, dependiendo a quien se busque notificar si se llegase a presentar un asalto en contra del usuario que hace uso de la aplicación. Los datos solicitados son: nombre completo, correo electrónico y número telefónico.

La consulta de contactos se emplea para visualizar los contactos registrados con anterioridad, estos se observan en forma de lista.

La edición de contactos se piensa para que el usuario actualice los datos de uno de los contactos registrados, por ejemplo, en caso de que un contacto haya cambiado su número telefónico o correo electrónico, ya que en varios casos los familiares y/o amigos cambian sus datos de contacto.

La configuración personalizada se crea como funcionalidad para que el usuario configure el texto y elija al contacto al cual se envía la notificación, el texto es completamente configurable.

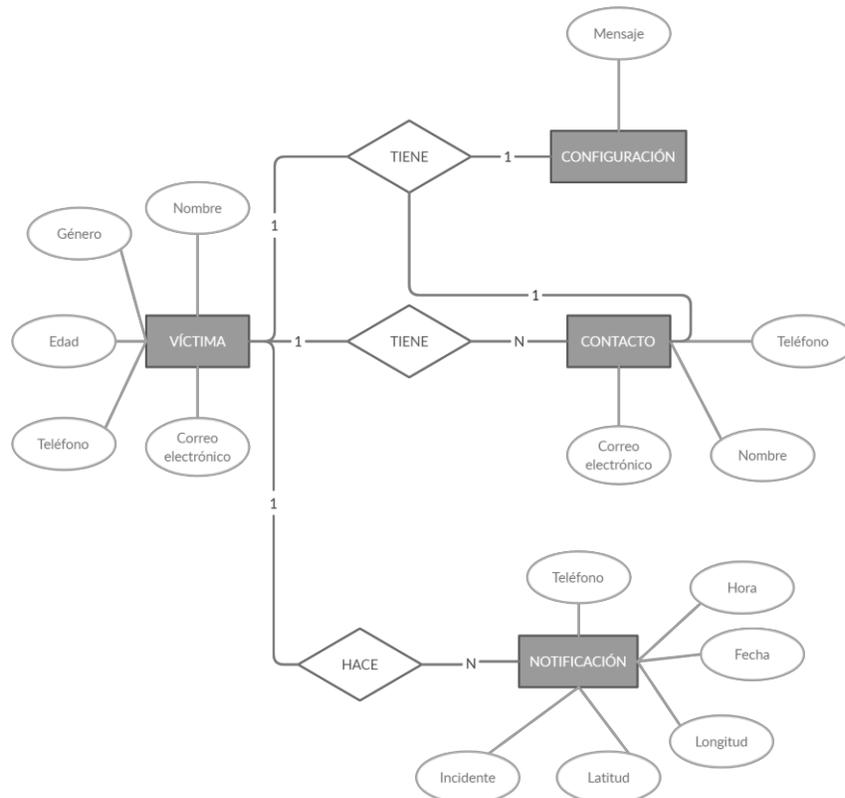


FIGURA 2.2 DIAGRAMA E-R

Para la tarea del envío de la notificación, se obtiene la configuración ingresada por el usuario en la configuración personalizada, una vez obtenidos estos datos, adicionalmente, se obtiene la latitud y longitud para crear el hipervínculo de *Google Maps*, la cual contiene la ubicación exacta del usuario, para notificar al contacto de emergencia y este pueda ubicar con mayor facilidad el lugar donde se encuentra el usuario.

Para poder guardar la información anteriormente mencionados se crea una base de datos y para modelarla se muestra en la Figura 2.2 el diagrama entidad relación.

Con esto definimos dos usuarios los cuales están interactuando con el sistema, dichos usuarios son:

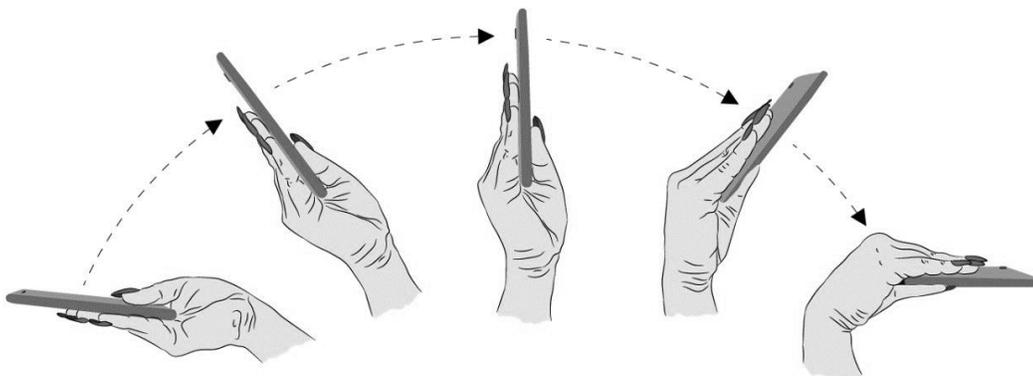
- Víctima o Agredido
- Contacto

Con el fin de lograr enviar las alarmas cuando se suscita el asalto en la vía pública se define un movimiento el cual tiene como fin entregar el teléfono con un movimiento de media luna de atrás hacia delante con la pantalla del teléfono celular hacia abajo.

Se define este movimiento porque tras observar en distintos lugares, como las personas entregan su teléfono celular se elige este movimiento debido a que es la forma más cotidiana en la que se entrega el teléfono, pero para evitar en la medida de lo posible que el usuario active de manera de inconsciente la interfaz gestual, se anexa que el teléfono sea entregado con la pantalla hacia abajo.

*IGEA* es el nombre que se adopta para referirse a la aplicación móvil desarrollada en esta tesis en futuras ocasiones, se elige este nombre por las siglas de **I**nterfaz **G**estual para el **E**nvío de **A**larmas.

En la Figura 2.3 se muestra gráficamente el movimiento propuesto en este trabajo, para activar el envío de una alarma



**FIGURA 2.3 MOVIMIENTO PROPUESTO**

Con la información de los casos de uso de la Figura 2.1, de la Figuras 2.4 a la 2.11 se muestra a mayor detalle las actividades involucradas en cada caso uso.

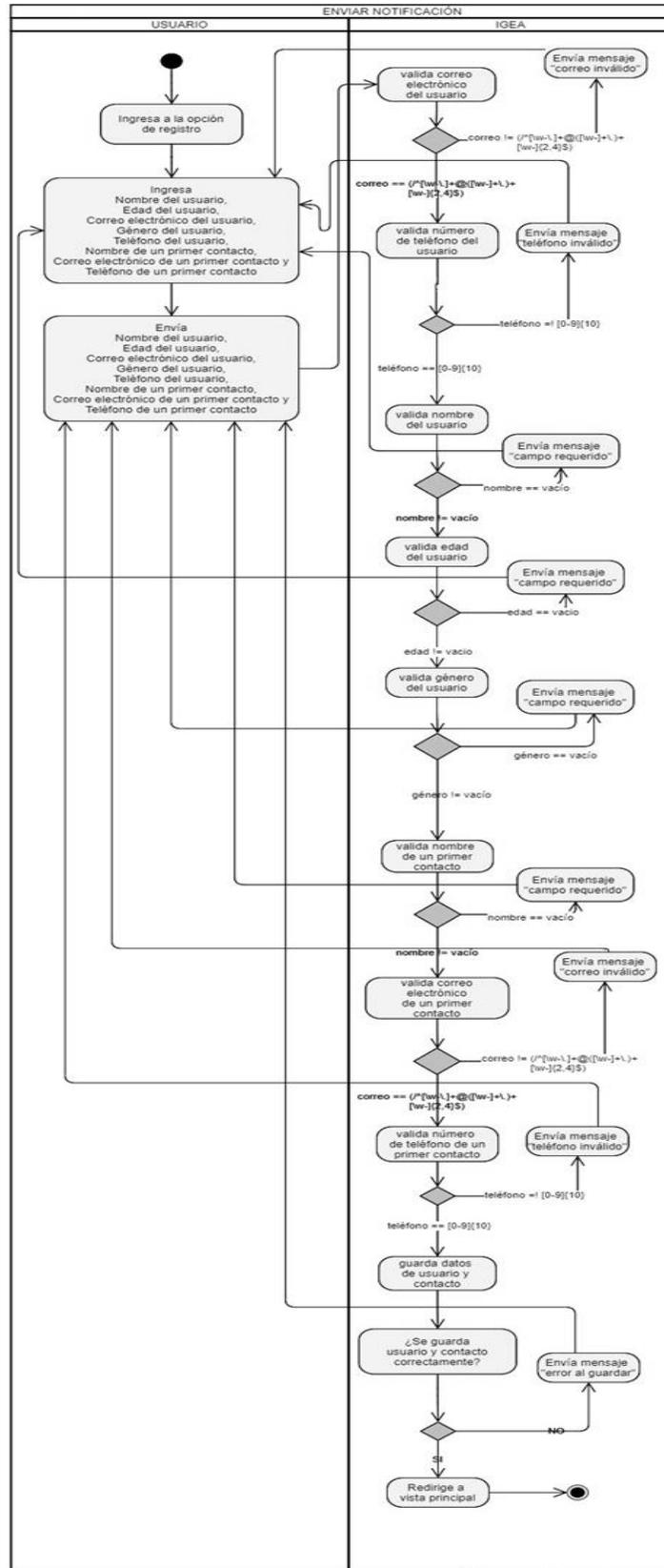


FIGURA 2.4 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE REGISTRAR

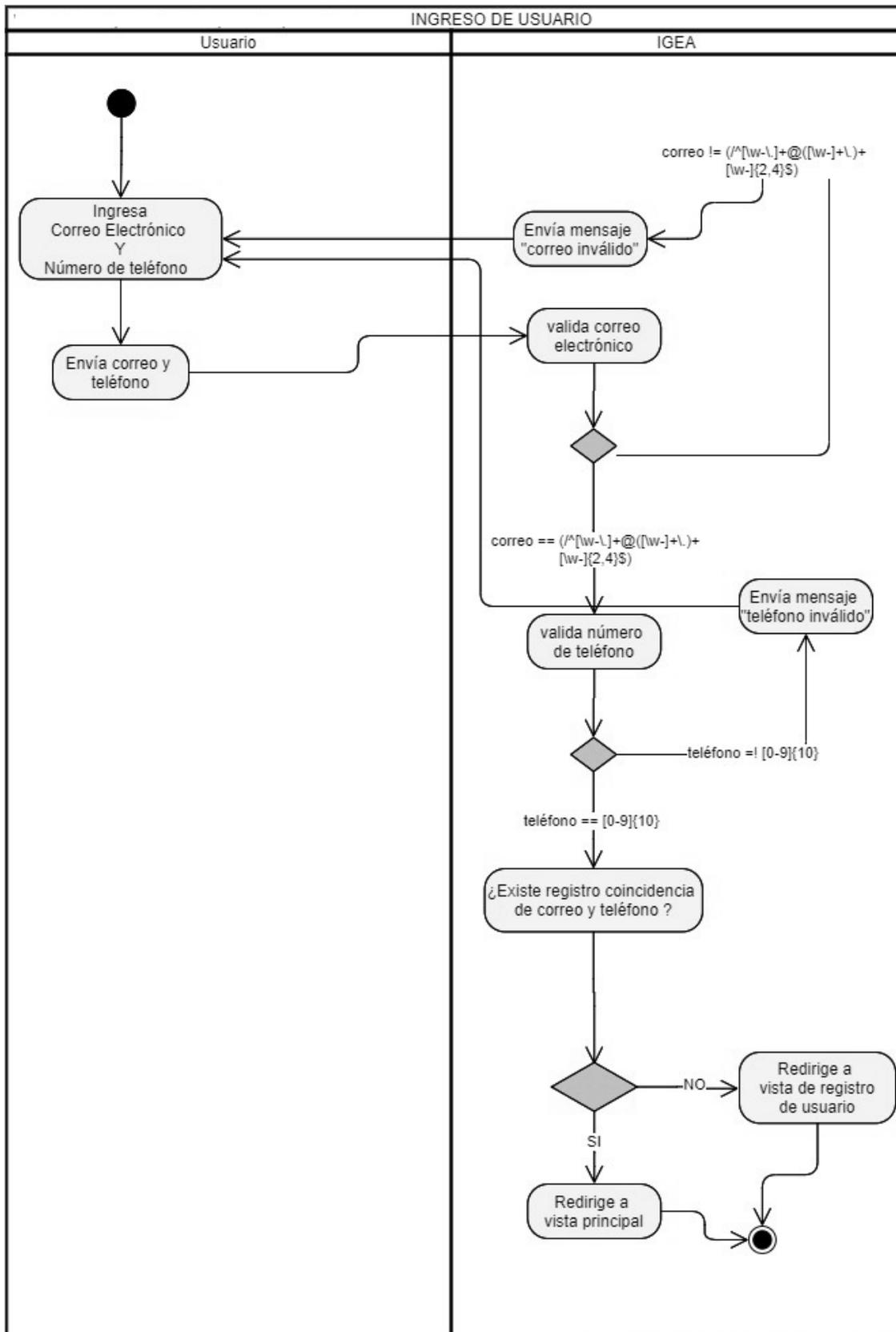


FIGURA 2.5 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE INGRESO

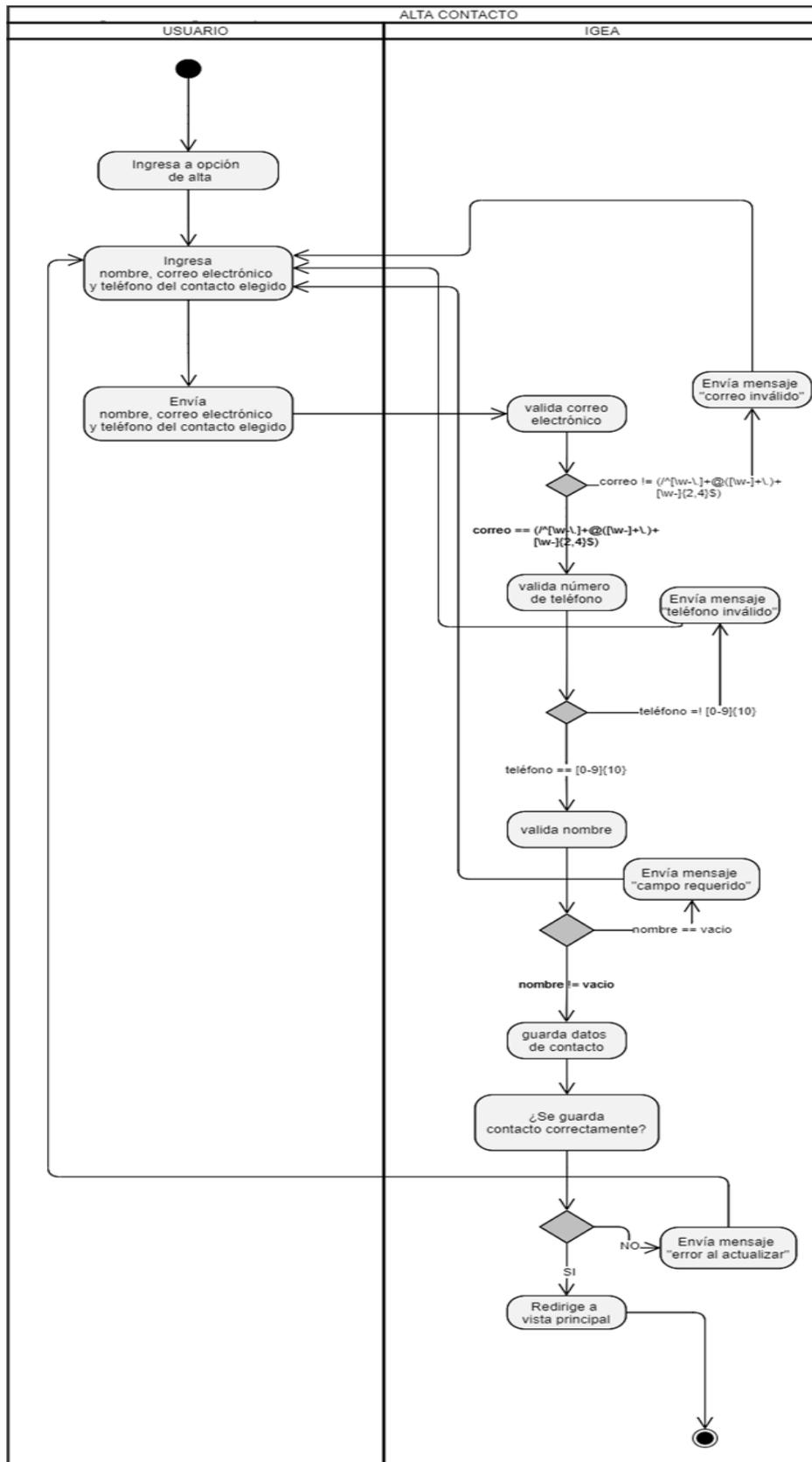


FIGURA 2.6 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE ALTA CONTACTO

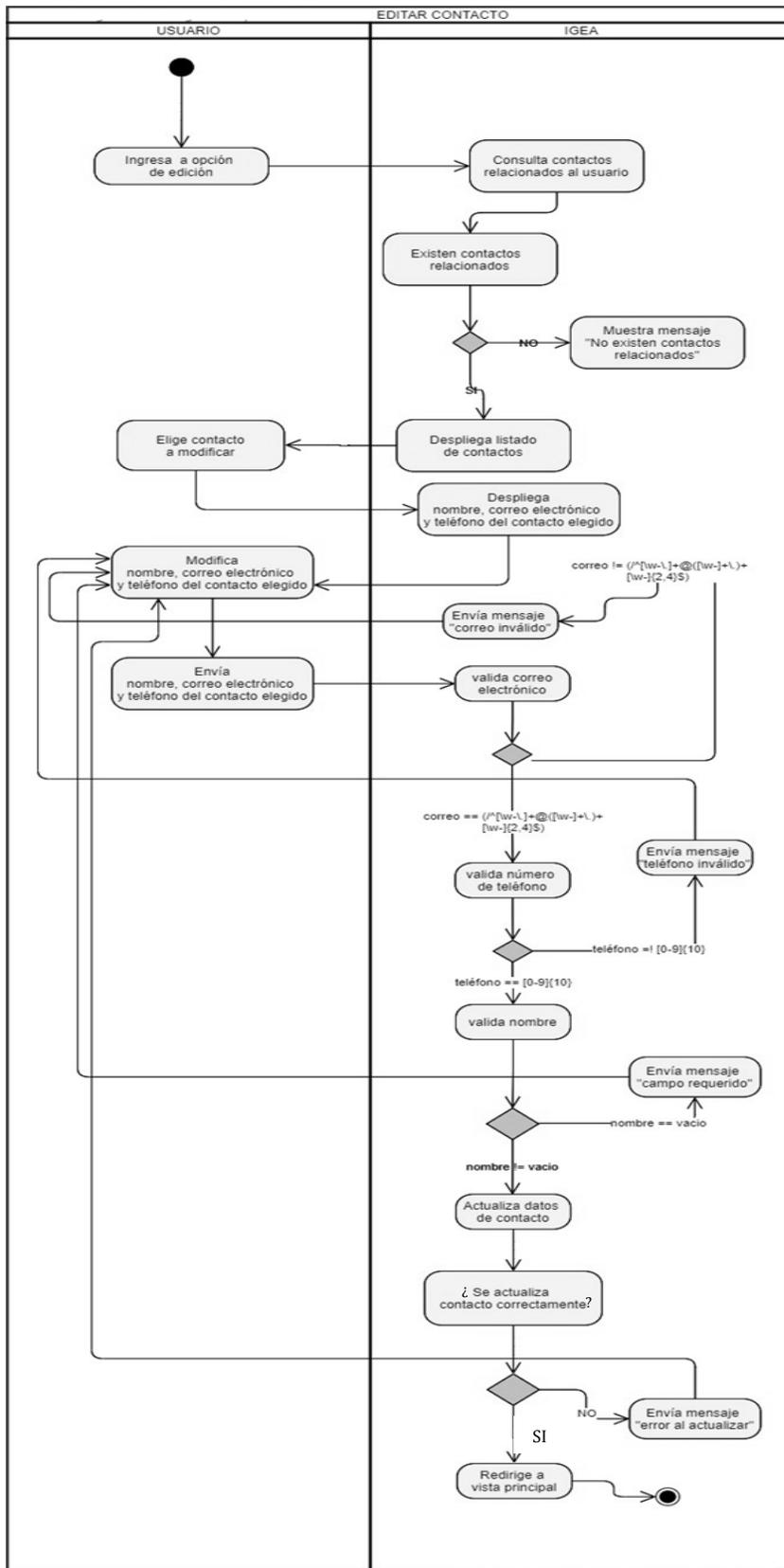


FIGURA 2.7 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE EDITAR CONTACTO

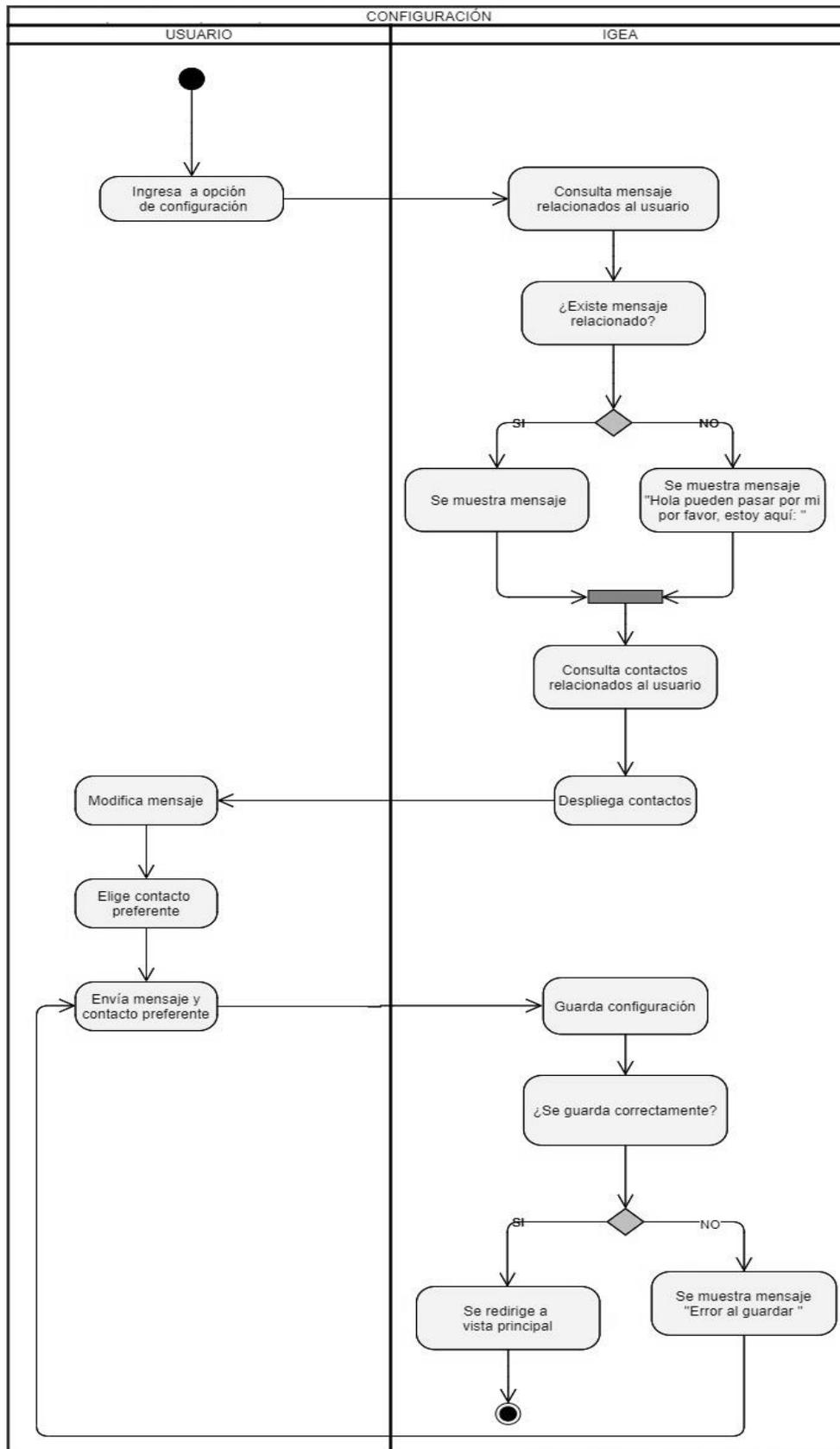


FIGURA 2.8 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE CONFIGURACIÓN DE MENSAJE

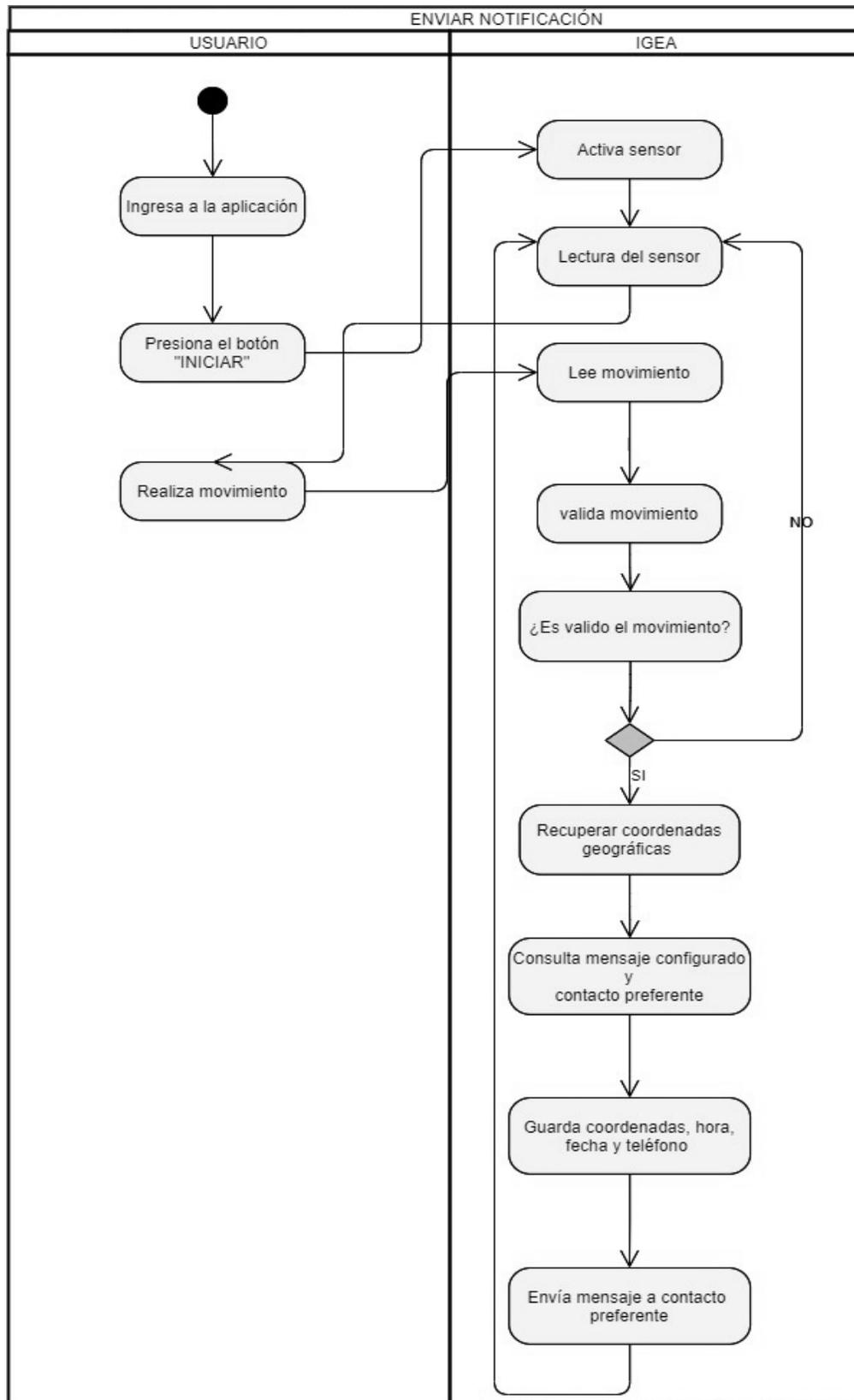


FIGURA 2.9 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE ENVIO DE NOTIFICACIÓN

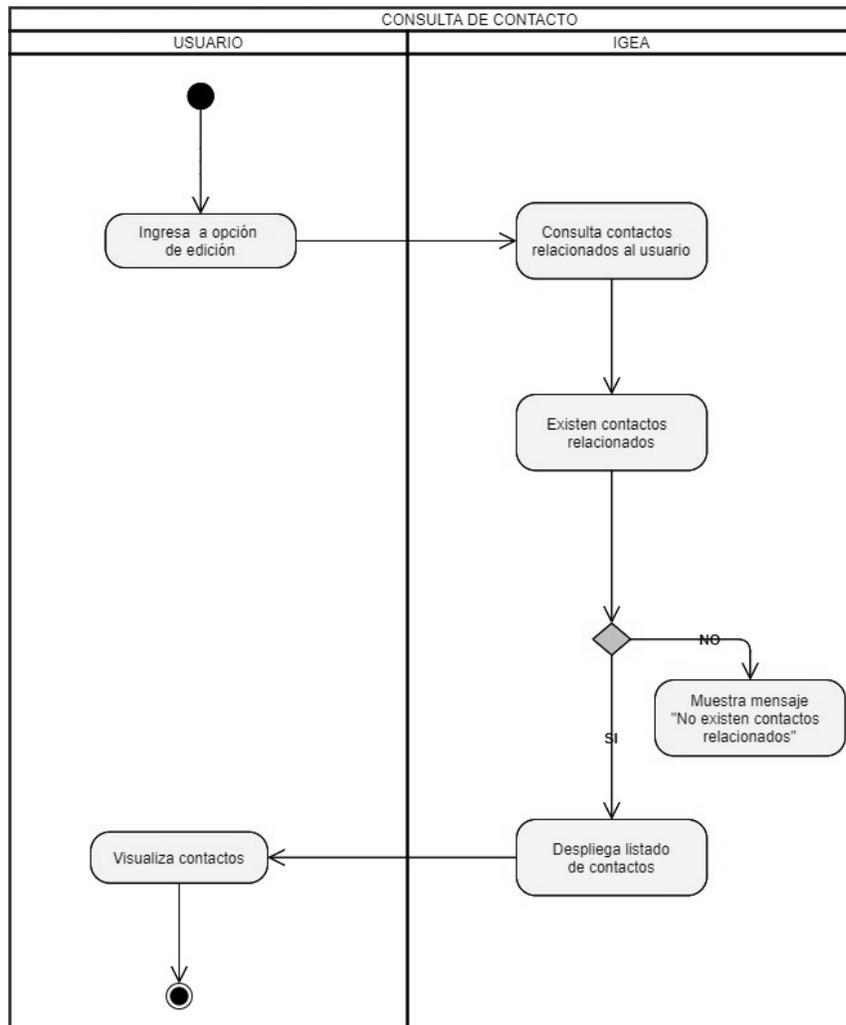


FIGURA 2.10 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE CONSULTA DE CONTACTO

Haciendo uso de la información de los casos de uso de la Figura 2.1, en la Figura 2.11 se muestran las actividades que se realizan en el envío de la notificación.

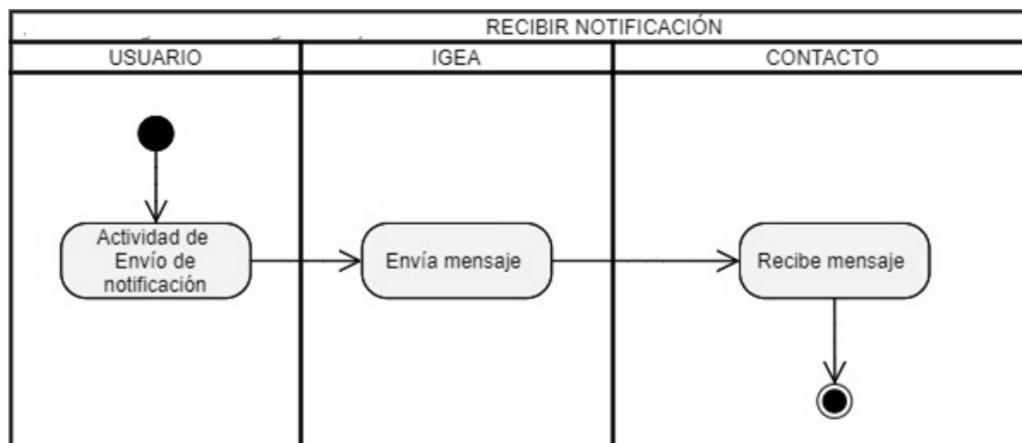


FIGURA 2.11 DIAGRAMA DE ACTIVIDADES DEL CASO DE USO DE RECIBIR NOTIFICACIÓN

El desarrollo se plantea para teléfonos celulares, ya que son objetos que llevamos con nosotros cotidianamente y porque cuentan con diversos sensores que facilitan la obtención de coordenadas geográficas y esto permite enviar la notificación.

Por accesibilidad se desarrolla la aplicación en *Android*, ya que, para realizar el desarrollo sobre este sistema, no se requiere ninguna clase de licencia porque este sistema operativo está basado en *software* libre.

Como sensor de movimiento se propone el acelerómetro porque es capaz de medir el desplazamiento de sus componentes para medir la aceleración, se hace esta propuesta por el rango de valores que se puede obtener de este sensor. Esta tecnología está incorporada en todos los teléfonos celulares, lo que permite implementar una aplicación *Android*. No se plantea el uso del giroscopio porque diversos modelos de teléfonos celulares aún no cuentan con este sensor de movimiento.

La primera actividad desarrollada de la aplicación *Android* es la creación de la base de datos, porque es el medio porque es el medio para almacenar los datos ingresados en la aplicación.

El servidor de base de datos se encuentra expuesto en una IP pública, con esto la conexión al servidor se puede hacer desde cualquier dispositivo que tenga salida a internet, el servidor de base de datos es prestado por parte de la empresa en la que actualmente estoy laborando. Este servidor es únicamente para las pruebas de la aplicación.

Para el diseño de la base de datos se piensa en cuatro tablas, las cuales son: víctima, contacto, configuración y notificación. Se muestra el diccionario de datos de cada tabla.

La tabla de víctima almacena los datos personales del usuario los cuales son: nombre, género, edad, correo electrónico y teléfono, los tipos de datos, de cada uno de los campos se muestran en la Tabla 2.3.

**TABLA 2.3 DICCIONARIO DE DATOS DE LA TABLA VÍCTIMA**

Campo	Tipo de dato	Función
<b>id</b>	BIGINT	Identificador único para el registro, este campo se emplea para realizar las relaciones con otras tablas.
<b>nombre</b>	VARCHAR (150)	Es el campo que almacena el nombre completo de los usuarios que van a utilizar la aplicación.
<b>edad</b>	INT	Es el campo que almacena la edad de los usuarios que van a utilizar la aplicación.
<b>género</b>	VARCHAR (1)	Es el campo que almacena el género de cada usuario registrado en la aplicación.
<b>correo</b>	VARCHAR (150)	Es el campo que almacena el correo electrónico de cada usuario que se registra en la aplicación.
<b>teléfono</b>	VARCHAR (10)	Es el campo que almacena el teléfono de cada usuario que se registra en la aplicación.

En la Tabla 2.4 se tienen los campos para la tabla contacto los cuales son: id, nombre, correo electrónico, teléfono, una llave foránea que hace referencia a la tabla de víctima. Adicionalmente se muestra el tipo de dato que tiene cada campo en la base de datos.

**TABLA 2.4 DICCIONARIO DE DATOS DE LA TABLA CONTACTO**

Campo	Tipo de dato	Función
<b>id</b>	BIGINT	Identificador único para el registro, este campo se emplea para realizar las relaciones con otras tablas.
<b>nombre</b>	VARCHAR (150)	Es el campo que almacena el nombre completo del contacto que registra el usuario en la aplicación.
<b>correo</b>	VARCHAR (150)	Es el campo que almacena el correo electrónico del contacto que registra el usuario en la aplicación.
<b>teléfono</b>	VARCHAR (10)	Es el campo que almacena el teléfono del contacto que registra el usuario en la aplicación.
<b>victima_id</b>	BIGINT	Es el campo que hace referencia al registro de víctima que registra el contacto.

La tabla configuración se piensa como una tabla donde se almacena el mensaje, el contacto al que el usuario o víctima desea notificar ante una situación de riesgo. Por lo que la definición de la misma queda como se muestra en la Tabla 2.5.

**TABLA 2.5 DICCIONARIO DE DATOS DE LA TABLA CONFIGURACIÓN**

Campo	Tipo de dato	Función
<b>id</b>	BIGINT	Identificador único para el registro, este campo se emplea para realizar las relaciones con otras tablas.
<b>Mensaje</b>	VARCHAR (100)	Es el campo que almacena el texto que se envía en la notificación.
<b>contacto_id</b>	BIGINT	Es el campo que indica el registro del contacto.
<b>victima_id</b>	BIGINT	Es el campo que indica el registro de la víctima.

Por último, la tabla de notificación es donde se almacena los datos de fecha, hora, latitud, longitud, número telefónico de la víctima e incidente. Su definición se muestra en la Tabla 2.6.

TABLA 2.6 DICCIONARIO DE DATOS DE LA TABLA NOTIFICACIÓN

Columna	Tipo de datos	Descripción del campo
<b>id</b>	BIGINT	Es el identificador único de cada registro que llega a la base de datos, este se genera automáticamente.
<b>fecha</b>	DATE	Es el campo donde se almacena la fecha con formato YYYY-MM-DD.
<b>Hora</b>	VARCHAR (8)	Es el campo donde se almacena la hora con formato HH:mm:ss.
<b>latitud</b>	VARCHAR (250)	Es el campo destinado a la latitud en la que se encuentra el dispositivo al momento de hacer el gesto propuesto.
<b>longitud</b>	VARCHAR (250)	Es el campo destinado a la longitud en la que se encuentra el dispositivo al momento de hacer el gesto propuesto.
<b>teléfono</b>	VARCHAR (8)	Es el campo destinado al teléfono del cual se envía la alarma.
<b>incidente</b>	VARCHAR (100)	Es el campo donde se almacena el tipo de incidente que reporta la víctima, aunque de momento dicho campo se guarda en vacío.

Por lo que el diagrama de clases de la base de datos resultante es el que se muestra en la Figura 2.12.

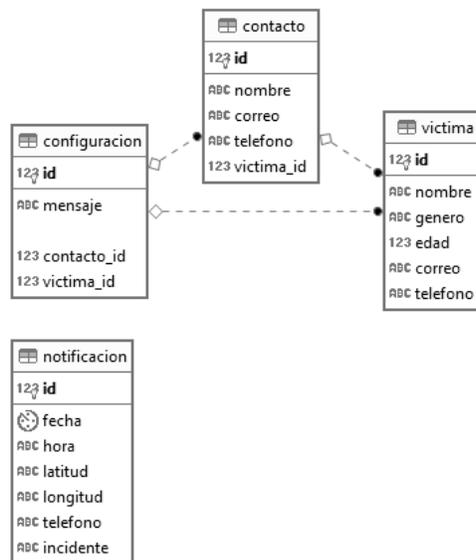


FIGURA 2.12 DIAGRAMA RELACIONAL DE BASE DE DATOS

Para realizar las operaciones de *CRUD (Create, Read, Update and Delete)* sobre estas tablas, se emplean *stored procedures*. Solo se mencionan los nombres de los *stored procedures* creados para la aplicación.

- spAltaUsuario
- spAltaContacto
- spConsultaContactos
- spConsultaUsuario
- spConsultaMensaje
- spActualizaContacto
- spAltaConfiguracion
- spAltaNotificacion

Para el desarrollo de la aplicación *Android*, se utiliza como *IDE Android Studio*, ya que este es el *IDE* oficial para el desarrollo de aplicaciones de este sistema operativo, aunque no es el único *IDE* que se puede usar para desarrollar aplicaciones *Android*.

Para el desarrollo de la aplicación, se emplean librerías adicionales a las que proporciona la aplicación *Android* por *default*, las cuales se describirá brevemente cuáles son y su función.

- a) *design*, dicha librería es la que permite dar el efecto de *tooltip* sobre un campo editable
- b) *cardview-v7*, esta librería es la que permite el diseño carta al momento de estar haciendo uso de *recyclerview*
- c) *recyclerview-v7*, es la librería que proporciona el *layout* de *recyclerview* para *Android*
- d) *jtds*, es la librería que permite la conexión a base de datos.

Para leer el movimiento propuesto se emplea el acelerómetro que es un sensor de movimiento con el que cuenta el teléfono celular, para obtener las coordenadas, se emplea el *GPS* del teléfono celular y para el envío de la alarma se ocupa *SMS* para poder notificar al contacto de emergencia, adicionalmente se requiere una conexión a internet para poder ingresar a la base de datos y almacenar los datos ingresados en la aplicación.

El proyecto de *Android* se divide en tres secciones:

- *assets*, es la carpeta donde se encuentra el archivo de configuración llamado *config.properties*, en el cual se encuentran los datos de conexión a la base de datos. Este archivo es pensado para que la conexión a base de datos sea configurable.
- *Java*, es la carpeta donde se encuentra el código *JAVA* que está asociado a los *activity* de la aplicación adicionalmente hay clases que contiene la gestión de la conexión a base de datos, el llamado a los *stored procedures*, etc.
- *res*, es la carpeta donde se encuentra todo lo relacionado con la gestión de elementos gráficos de la aplicación, pero lo que cabe resaltar de esta sección son los archivos *strings.xml*, *colors.xml* y *styles.xml*, ya que, en estos archivos se encuentran los estilos, textos y colores que se muestran en toda la aplicación y todos los valores que se encuentra en estos archivos son editables a nivel código, para la personalización de la aplicación.

En la sección de *Java* las clases que tiene la funcionalidad más importante de la aplicación son:

- *AlarmReceiver*, es la clase encargada de hacer que la aplicación se ejecute en segundo plano, siempre y cuando la aplicación no sea cerrada completamente, adicionalmente ejecuta la clase *SchedulingService*.
- *SchedulingService* es una clase servicio, sus funciones principales son: leer el sensor de movimiento, validar el gesto propuesto, obtener las coordenadas geográficas por medio del llamado a la clase *Localizacion* y enviar la notificación si el gesto ejecutado es el correcto.

- Nota, para versiones superiores a *Android 5*, al momento de iniciar la aplicación esta solicitará permisos para el uso del GPS y el envío de *SMS*, esto por parte de las mejoras de seguridad que existen desde *Android 6.0*.
- *Localizacion*, es la clase que recupera los valores de latitud y longitud que son proporcionados por el GPS.

## 2.4 PROTOTIPO FUNCIONAL

En esta sección se muestra el prototipo inicial y el primer refinamiento de *IGEA* a nivel interfaz gráfica de usuario.

### 2.4.1 PROTOTIPO INICIAL

En esta sección se muestran las interfaces gráficas de usuario diseñadas para el prototipo inicial de *IGEA*, adicionalmente se describe la función de cada estas.

En la Figura 2.13 se muestra la interfaz principal de *IGEA*, la cual contiene dos botones, el primer botón de iniciar y el segundo de detener. El primer botón sirve para iniciar el servicio de lectura del acelerómetro, adicionalmente mantiene la lectura del acelerómetro en segundo plano y en caso de ejecutarse el movimiento correcto, dicho servicio obtiene las coordenadas, el número telefónico del contacto preferente y, por último, se guarda toda la información recabada en base de datos.

Aun cuando la aplicación no se encuentre en primer plano realiza toda la funcionalidad anteriormente descrita.

El segundo -botón tiene la funcionalidad de detener el servicio de lectura del acelerómetro.



FIGURA 2.13 *IGEA* - PRINCIPAL

### 2.4.2 PRIMER REFINAMIENTO

En esta sección se muestran las interfaces gráficas de usuario diseñadas para el primer refinamiento del prototipo de *IGEA*, adicionalmente se describe la funcionalidad a detalle de cada una de las interfaces y como interactúan con el usuario a nivel gráfico.

En la Figura 2.14 y 2.15 son las interfaces diseñadas para versiones de *Android* 6 o superiores debido a que, a partir de estas versiones, se requiere la autorización por parte del usuario para poder emplear ciertos recursos del dispositivo móvil. En este caso se solicitan permisos para el uso del *GPS* y para el envío de mensajes.

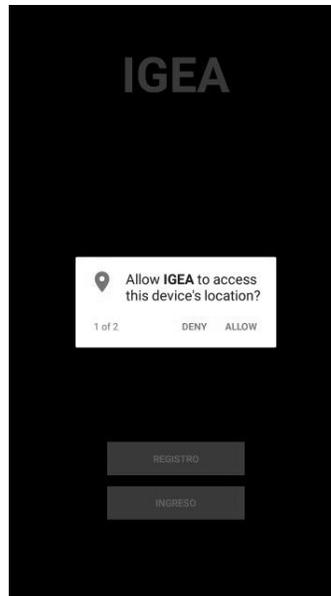


FIGURA 2.14 *IGEA* - SOLITUD DE PERMISOS PARA USO DEL GPS

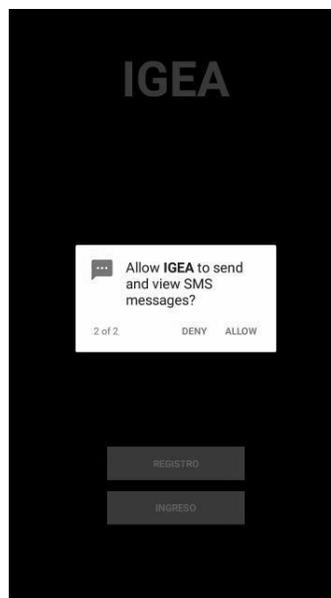
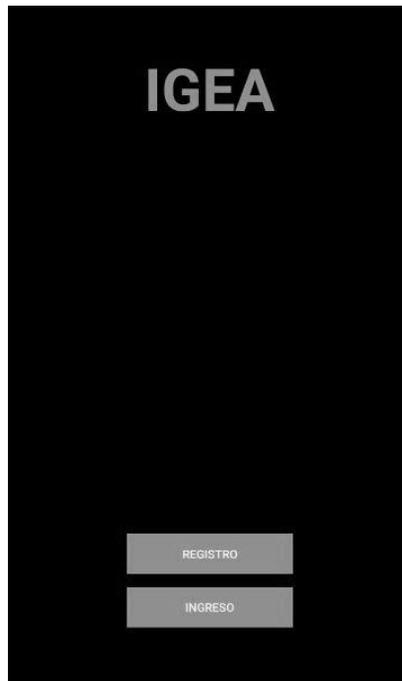


FIGURA 2.15 *IGEA* - SOLITUD DE PERMISOS PARA EL ENVÍO DE MENSAJES SMS DESDE LA APLICACIÓN

Tras haber autorizado ambos permisos se visualiza la Figura 2.16 o en caso de ser una versión inferior a *Android 6.0* no se muestra ninguna de las dos solicitudes de permisos anteriores, por lo que se visualiza la Figura 2.16.

En caso de no aceptar los permisos, la aplicación solicita nuevamente los permisos al ingresar a la aplicación, la solicitud de permisos se muestra hasta que sean aceptados todos los permisos.

En la Figura 2.16 se observan dos botones, el primer botón nos redirige a la Figura 2.17 la cual es la *GUI* de registro de usuario; esta opción es para los usuarios que van a usar la aplicación por primera vez, el segundo botón nos redirigirá a la Figura 2.18 la cual representa la *GUI* de ingreso a *IGEA*; esta opción es para los usuarios previamente registrados.



**FIGURA 2.16 IGEA - INICIO**

En la Figura 2.17 se muestra la vista de registro de *IGEA* dentro de esta se muestran los campos necesarios para el registro del usuario, cada campo solicitado cuenta con su propia validación para conservar la integridad de la base de datos.

La primera sección de la información solicitada corresponde a los datos del usuario, mientras que la segunda parte de los datos corresponde a un primer contacto preferente para que la aplicación pueda enviar la notificación a dicho contacto, sin embargo, para la sección de activación de la interfaz gestual es necesario ingresar un mensaje personalizado para habilitar el botón que inicia la interfaz gestual.

11:22

### REGISTRO

DATOS PERSONALES

Nombre \_\_\_\_\_

Edad \_\_\_\_\_

Correo Electrónico \_\_\_\_\_

Masculino  Femenino

Teléfono \_\_\_\_\_

DATOS DE CONTACTO

Nombre \_\_\_\_\_

Correo Electrónico \_\_\_\_\_

Teléfono \_\_\_\_\_

GUARDAR

FIGURA 2.17 IGEA - REGISTRO DE USUARIO

En la Figura 2.18 se muestra la vista de ingreso de la aplicación, en esta se solicita el correo y número de teléfono celular con el que se dio de alta el usuario, todos los campos solicitados en esta vista cuentan con su validación propia, tras ingresar los datos estos son validados a nivel base de datos. En caso de encontrar una coincidencia, la aplicación nos envía directamente a la vista de inicio de *IGEA*, en caso contrario se muestra un mensaje informativo que no existe un registro con los datos ingresados.

11:22

### INGRESO

Correo Electrónico \_\_\_\_\_

Teléfono \_\_\_\_\_

ENTRAR

FIGURA 2.18 IGEA - INGRESO

En la Figura 2.19 se muestra la vista principal de *IGEA*, esta vista cuenta con varios botones que tienen diferentes funcionalidades.

Para el botón con el icono del engranaje, nos redirige a la Figura 2.20, esta figura corresponde al apartado de configuración del texto y elección del contacto preferente.

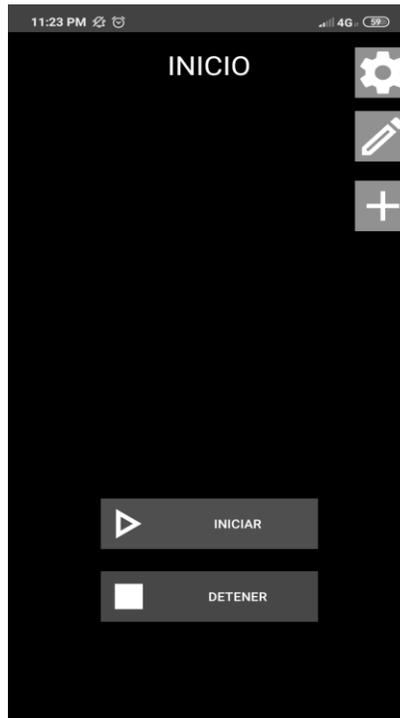
El botón con el icono de lápiz, tiene la funcionalidad de redirigirnos a la Figura 2.22, la cual muestra los contactos que se tengan registrados.

En cuanto al botón con el icono con el signo de más, nos envía a la Figura 2.24, la cual se encarga de registrar nuevos contactos de emergencia.

El botón de iniciar, inicia el servicio de lectura del acelerómetro, adicionalmente mantiene la lectura del acelerómetro en segundo plano y en caso de ejecutarse el movimiento correcto, dicho servicio obtiene las coordenadas, el número telefónico del contacto preferente y por último, envía el mensaje de notificación con el texto configurado y una liga de *Google Maps* con las coordenadas geográficas exactas de donde se encuentra el usuario.

Aun cuando la aplicación no se encuentre en primer plano realiza toda la funcionalidad anteriormente descrita. Sin embargo, este botón se encuentra deshabilitado al ingresar por primera vez a *IGEA*, tras configurar el texto y elegir al contacto dicho botón se habilita.

El botón de detener tiene la funcionalidad de parar el servicio de lectura del acelerómetro.



**FIGURA 2.19 IGEA - PRINCIPAL**

En la Figura 2.20 se muestra la vista de configuración, la funcionalidad de esta vista es establecer el texto a enviar y elegir al contacto preferente al cual se le envía la notificación.

En la Figura 2.21 dentro de esta vista, se muestra un listado de contactos, dicho listado depende de los contactos que el usuario haya registrado en *IGEA*.

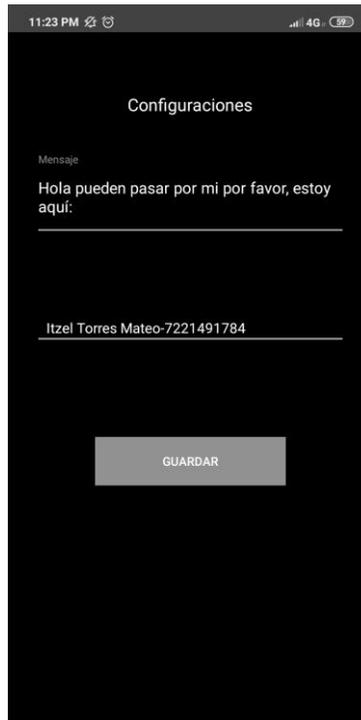


FIGURA 2.20 *IGEA* - CONFIGURACIÓN DE LA APLICACIÓN

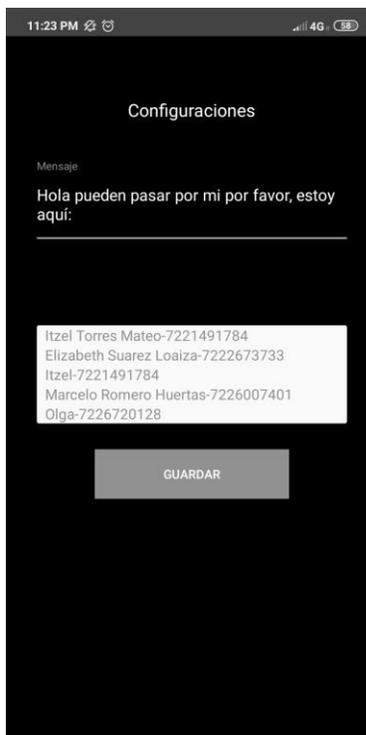
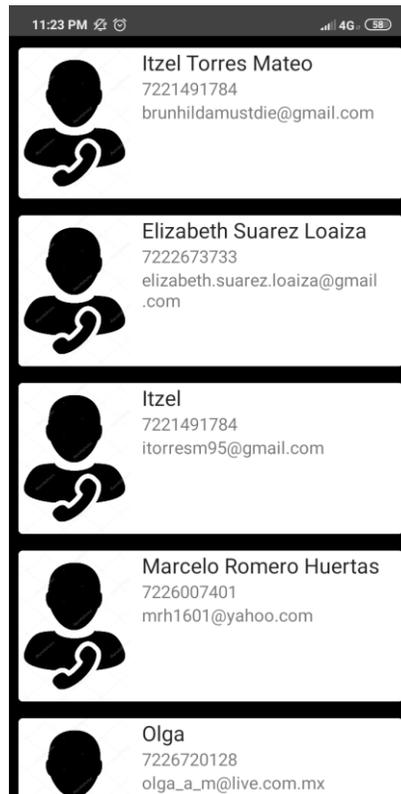


FIGURA 2.21 *IGEA* - CONFIGURACIÓN DE LA APLICACIÓN

Para la Figura 2.22, se muestra un listado de los contactos registrados por el usuario desde *IGEA*. Al elegir alguna de las fichas de contacto este enviará a la vista de edición de contacto, de acuerdo con la ficha elegida en la interfaz.

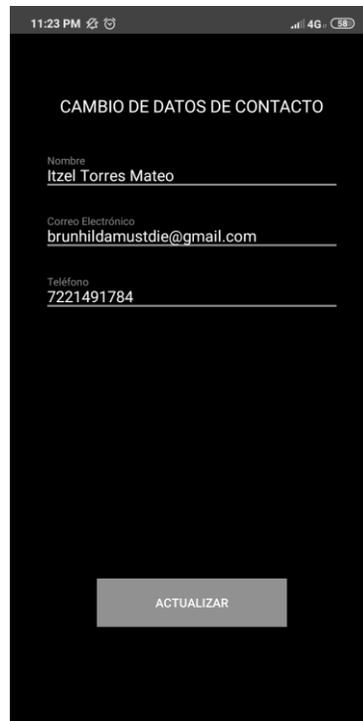
En la Figura 2.23 se muestran los datos del contacto que fue elegido en la vista de consulta de contacto en este caso se está refiriendo a la Figura 2.22.

La funcionalidad de la vista en la Figura 2.23 es actualizar los datos del contacto elegido por el usuario.



**FIGURA 2.22 IGEA - LISTADO DE CONTACTOS DE LA APLICACIÓN**

En la Figura 2.24 corresponde a la vista de alta de contacto de la aplicación, dentro de esta GUI se solicitan datos como nombre, correo electrónico y número telefónico para guardarlo en base de datos. Para que dicho contacto pueda ser empleado más adelante como un posible contacto preferente, el cual puede recibir la notificación. Adicionalmente, cada campo cuenta con su propia validación para poder asegurar la integridad de los datos que haya ingresado el usuario.



**FIGURA 2.23 IGEA - EDICIÓN DE DATOS DE CONTACTO**



**FIGURA 2.24 IGEA - ALTA DE DATOS DE CONTACTO**

En el capítulo III, se realizan las pruebas del primer prototipo de la interfaz gestual desarrollada.

En el capítulo III, también se incluye el análisis de los resultados obtenidos al realizar las pruebas con usuarios.

## CAPÍTULO III. EVALUACIÓN DE LA APLICACIÓN MÓVIL

---

*En este capítulo se describen las pruebas realizadas con la interfaz y cuáles fueron los resultados después de haber hecho la experimentación.*

### 3.1 EVALUACIÓN DEL PROTOTIPO INICIAL

#### Pruebas basadas en el usuario

Considerando 45 usuarios para la prueba del prototipo inicial de *IGEA*, cada participante cumple con las características:

- Tenga entre 15 a 50 años cumplidos.
- Cuente por lo menos con una de sus extremidades superiores completas.
- Cuente con un dispositivo con red de telefonía.
- Cuente con red de datos disponible.
- Cuente con un dispositivo móvil con versión de Android 5.0 o superior con GPS y saldo suficiente para poder realizar el envío de la notificación (Preferentemente).
- Escolaridad mínima de secundaria.
- Habilidades avanzadas en el uso de teléfonos inteligentes.

#### Escenario de experimentación

Como fue planteado en el objetivo general de esta tesis, la medición de la exactitud debe realizarse cuando el usuario se encuentra en situaciones de riesgo. Por esta razón, los escenarios planteados para esta tesis se consideran en la vía de tres colonias del Estado de México: Santa Cruz Atzacapotzaltongo y Universidad del municipio de Toluca, y El Calvario del municipio de Zinacantepec, para simular el escenario de robo a peatones. Esto se plantea de esta forma, ya que la mayoría de los usuarios que participaron en las pruebas tuvieron miedo de salir a los lugares plantados originalmente. Pero esta prueba se realiza empleado los datos móviles de sus teléfonos celulares.

#### Métricas de usabilidad

En la Tabla 3.1 se observan las métricas de usabilidad que se emplean para medir la usabilidad del primer prototipo de *IGEA*.

**TABLA 3.1 MÉTRICAS DE USABILIDAD DEL PRIMER PROTOTIPO**

Indicadores de efectividad	
Del usuario	<ul style="list-style-type: none"><li>• Facilidad en el manejo de la aplicación</li><li>• Problemas en el manejo de la aplicación</li></ul>
De la aplicación	<ul style="list-style-type: none"><li>• Número de operaciones exitosas</li><li>• Número de operaciones fallidas</li></ul>

Para evaluar Número de operaciones exitosas y Número de operaciones fallidas, se realiza un conteo del número de mensajes recibidos una vez hecho el gesto que envía la notificación.

Para evaluar Facilidad de uso de la aplicación y Problemas en el manejo de la aplicación, se toma en cuenta el grado satisfacción que tienen los usuarios al estar utilizando la aplicación gestual.

### **Mejoras al prototipo**

Tras concluir con las pruebas del primer prototipo de *IGEA*, se encuentran las siguientes mejoras:

- Notificar de la situación de riesgo a un tercer para completar el canal de comunicación.
- Cambiar el diseño de la interfaz gráfica de usuario ya que esta es poco agradable para el usuario.
- Agregar módulos para agregar, consultar, editar contactos.

## **3.2 EVALUACIÓN DEL PRIMER REFINAMIENTO DEL PROTOTIPO**

En esta sección se describe la evaluación del primer refinamiento del prototipo de la aplicación móvil desarrollada en esta tesis.

### **3.2.1 PRUEBAS BASADAS EN EL USUARIO**

Lo que se busca con las pruebas es medir la facilidad que tiene el usuario utilizar la aplicación, medir el tiempo que le toma al usuario realizar todas las acciones que se pueden realizar en la misma. Las cuales son: el registro del usuario y un primer contacto de emergencia, configuración del mensaje y del contacto al cual se le enviará la notificación, operaciones de alta, consulta y edición de los contactos de emergencia registrados en la aplicación y la activación de la interfaz gestual.

También se busca medir la eficiencia y eficacia de la aplicación al momento de realizar todas las tareas para las que fue diseñada.

#### **3.2.1.1 SELECCIÓN DE USUARIOS**

Se consideran 15 participantes para la prueba de la aplicación, cada participante cumple con las siguientes características:

- Tenga entre 15 a 50 años cumplidos.
- Cuente por lo menos con una de sus extremidades superiores completas.
- Cuente con un dispositivo con red de telefonía.
- Cuente con red de datos disponible.
- Cuente con un dispositivo móvil con versión de *Android* 5.0 o superior con GPS y saldo suficiente para poder realizar el envío de la notificación (Preferentemente).
- Escolaridad mínima de secundaria.
- Habilidades avanzadas en el uso de teléfonos inteligentes.

La razón por la cual se solicitaron estas características fue porque:

- El público de este rango de edades son los más propensos a sufrir un asalto en la vialidad pública.
- Para poder realizar la acción principal de la interfaz gestual, se requiere que se realice un movimiento con la mano, por lo que el público que no cuenta con al menos una de sus extremidades superiores no podía ser candidato para estas pruebas.

- Preferentemente solicito al usuario que contará con teléfono celular con *Android 5.0* como versión mínima, sin embargo, si no contaba con un teléfono con estas características, se le proporcionaba uno para la prueba. Con única condición que lo devolviera al final de la prueba.
- Y el último punto fue pensado para reducir el tiempo de la prueba, pero no es obligatorio cumplir con este punto.

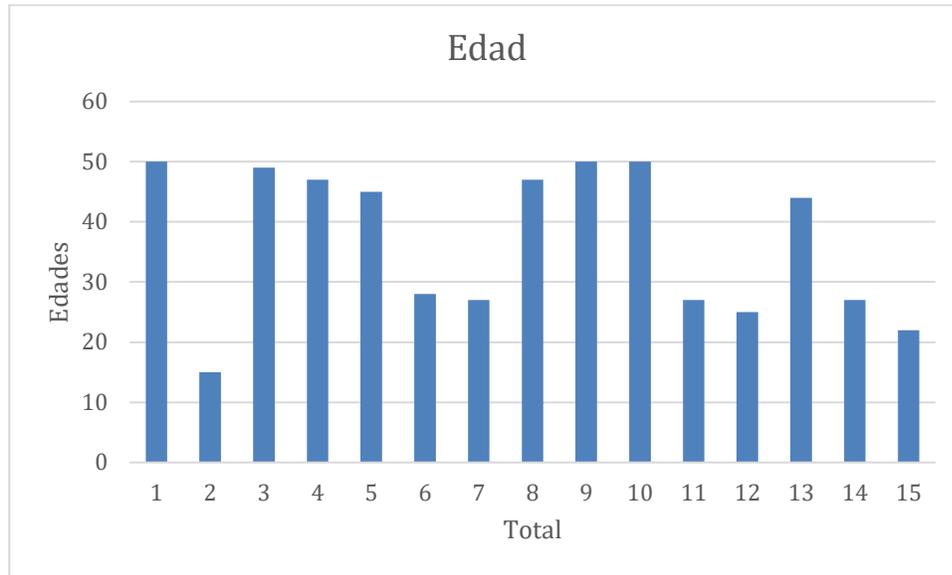
Tras haber definido las características que deben tener los usuarios. En la Tabla 3.2 se muestran diversos datos de los usuarios los cuales son: nombre, edad, escolaridad y género de cada de los usuarios que apoyaron en las pruebas de usabilidad de la interfaz gestual. Para proteger la privacidad de los usuarios no se muestra su nombre real.

**TABLA 3.2 MUESTRAS USADA PARA LAS PRUEBAS DE USABILIDAD**

Nombre	Edad	Escolaridad	Sexo
<b>Usuario 1</b>	50	Maestría	Femenino
<b>Usuario 2</b>	15	Secundaria	Masculino
<b>Usuario 3</b>	49	Maestría	Femenino
<b>Usuario 4</b>	47	Maestría	Masculino
<b>Usuario 5</b>	45	Maestría	Masculino
<b>Usuario 6</b>	28	Licenciatura	Femenino
<b>Usuario 7</b>	27	Licenciatura	Femenino
<b>Usuario 8</b>	47	Maestría	Femenino
<b>Usuario 9</b>	50	Licenciatura	Masculino
<b>Usuario 10</b>	50	Licenciatura	Femenino
<b>Usuario 11</b>	27	Licenciatura	Masculino
<b>Usuario 12</b>	25	Licenciatura	Femenino
<b>Usuario 13</b>	44	Secundaria	Femenino
<b>Usuario 14</b>	27	Preparatoria	Masculino
<b>Usuario 15</b>	22	Secundaria	Masculino

Con la información recabada de los usuarios se realiza un análisis cuantitativo de las características proporcionadas, las características consideradas son: edad, escolaridad y sexo. Lo que se busca con este análisis es ver qué sectores de la población la aplicación de interfaz gestual tiene una mayor influencia.

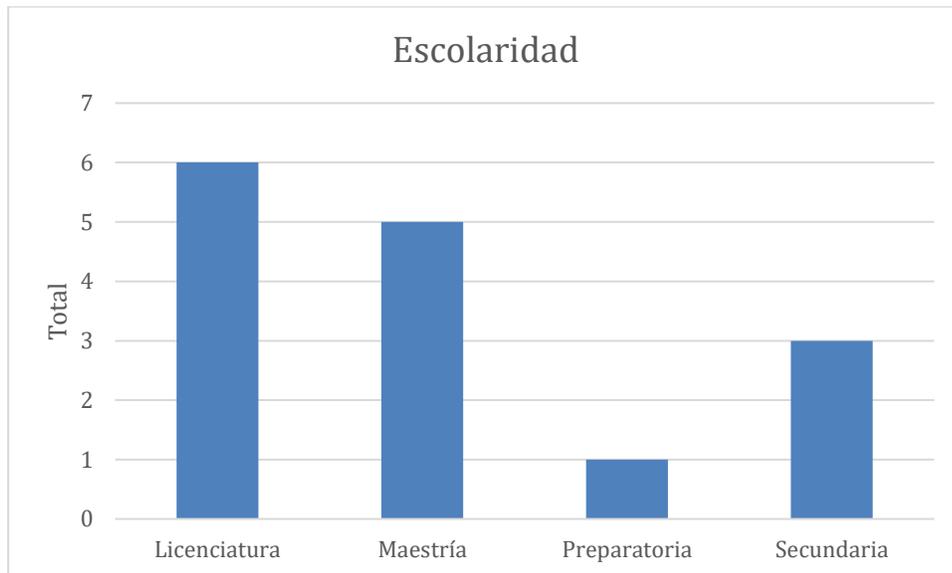
Se toma la característica de la edad de la Tabla 3.1, se puede observar que las edades con mayor concurrencia son edades de veintisiete y cincuenta años. Así como se muestra en la Gráfica 3.1.



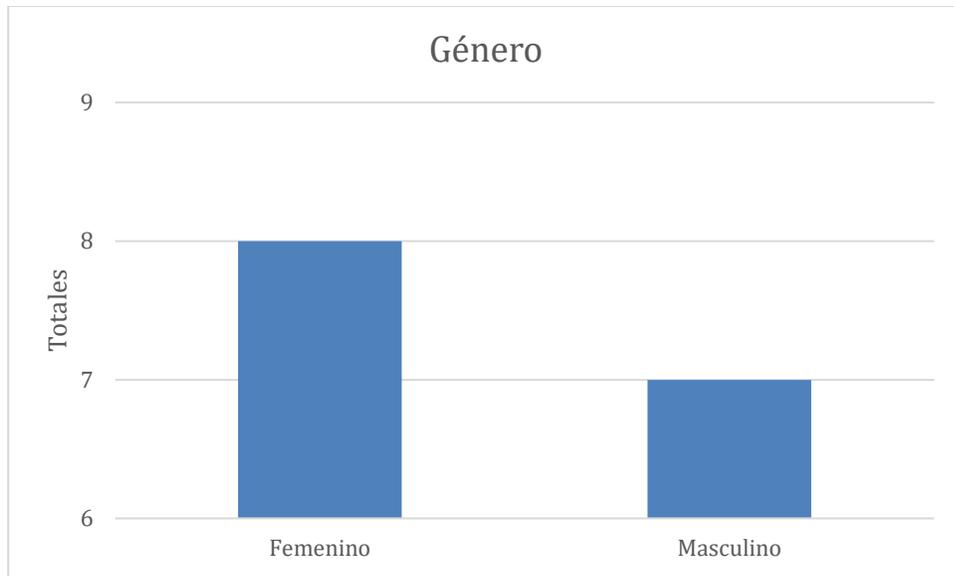
GRÁFICA 3.1 GRÁFICA DE EDADES

De la misma manera se realiza el análisis de los datos con respecto a la escolaridad de cada usuario que aplica en la prueba, y podemos ver que el mayor uso es por usuarios que cuentan con licenciatura y maestría. Así como se muestra en la Gráfica 3.2.

GRÁFICA 3.2 GRÁFICA DE ESCOLARIDAD



De la misma manera el análisis de los datos con respecto al género de cada usuario que aplica en la prueba, donde el mayor uso es por aquellos usuarios con sexo femenino. Así como se muestra en la Gráfica 3.3.



GRÁFICA 3.3 GRÁFICA DE GÉNERO

### 3.2.1.2 TAREAS A REALIZAR

Se retoman las tres tareas definidas anteriormente: instalación, configuración de la aplicación móvil y envío de alarma. Con el fin de evaluar la aplicación con respecto a las tareas que desempeña, adicionalmente se proporciona una descripción de como se realizan estas tareas.

La tarea de instalación involucra la ejecución de dos acciones: descarga de la aplicación e instalación de la aplicación. Por otro lado, la tarea de configuración incluye dos acciones: registro del usuario en la aplicación y registro del contacto de emergencia, configuración del texto y elección del contacto de emergencia. Mientras que el envío de alarma involucra dos acciones: activar la interfaz gestual y realizar el movimiento para activar la interfaz gestual.

#### Tarea 1: Instalación de la aplicación

La tarea de instalación de la aplicación contiene dos acciones principales las cuales son: descarga de la aplicación e instalación de la aplicación, a continuación, se describe detalladamente como se realizaron las acciones, para poder cumplir ambas acciones.

- 1) Para el proceso de descarga de la aplicación gestual, se debe entrar al siguiente hipervínculo [https://drive.google.com/file/d/1NsvZA8rEbBXHA-cvUN0kobp\\_6jpMnBk1/view](https://drive.google.com/file/d/1NsvZA8rEbBXHA-cvUN0kobp_6jpMnBk1/view)
- 2) Para el proceso de instalación de la aplicación gestual se siguieron los pasos siguientes
  - i) Para hacer esto debe ingresar a la opción de "Configuración" del dispositivo móvil una vez ahí se debe buscar la opción "Acerca del teléfono". Como se puede ver la Figura 3.3.

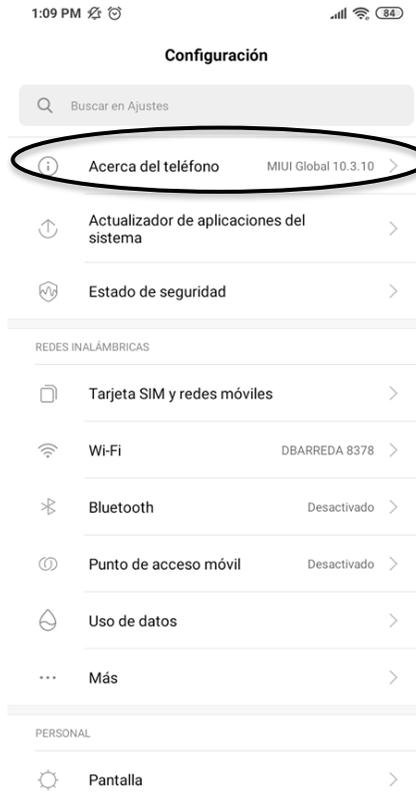


FIGURA 3.3 GUI - MÓDULO ACERCA DEL TELÉFONO EN XIAOMI REDMI 7

- ii) Tras ingresar a esa opción se debe buscar la opción “Versión de *Android*” y en el apartado se podrá ver la versión del sistema operativo, como se muestra en la Figura 3.4.

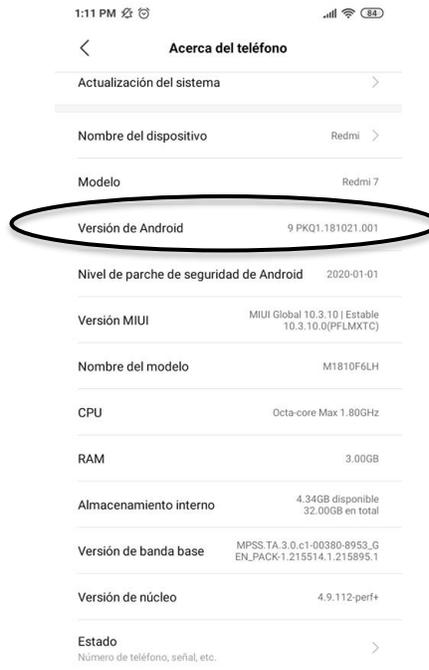


FIGURA 3.4 GUI - DETALLE DE MÓDULO ACERCA DEL TELÉFONO EN XIAOMI REDMI 7

- iii) En caso de ser una versión igual o mayor a 6 y menor a 7 ir al punto b.
- iv) En caso de ser una versión superior a 7 ir al punto c.
- b) Lo primero que se tiene que hacer es activar los orígenes desconocidos. Como su nombre bien indica, el objetivo es poner instalar aplicaciones que provienen de fuera de la Google Play. En la Figura 3.5 se muestra la forma para activar dicha opción.

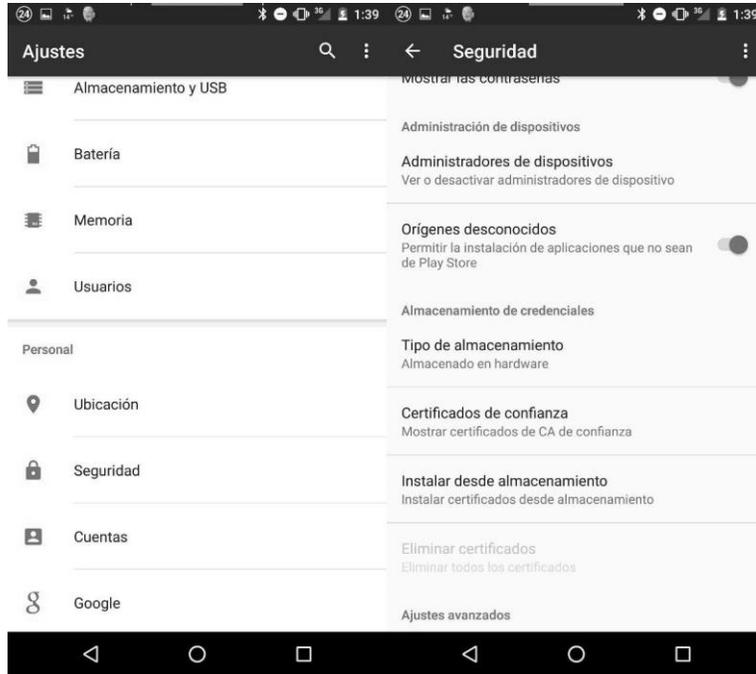


FIGURA 3.5 GUI - ACERCA DEL TELÉFONO EN DISPOSITIVO MÓVIL GENÉRICO

- c) Para versiones iguales o superiores a Android 7 la opción de orígenes desconocidos ha desaparecido. Sin embargo, se puede llevar a cabo la instalación de la aplicación realizado los siguientes pasos como ejemplo.
  - i) Descarga el APK o APKs externos (*Google Chrome, Telegram, WhatsApp, etc.*, para este ejemplo se empleará *Google Chrome* como medio de descarga.).
  - ii) Al finalizar la descarga, se debe instalar la aplicación. Al iniciar la instalación el sistema operativo mostrará un mensaje informativo como se muestra en la Figura 3.6.

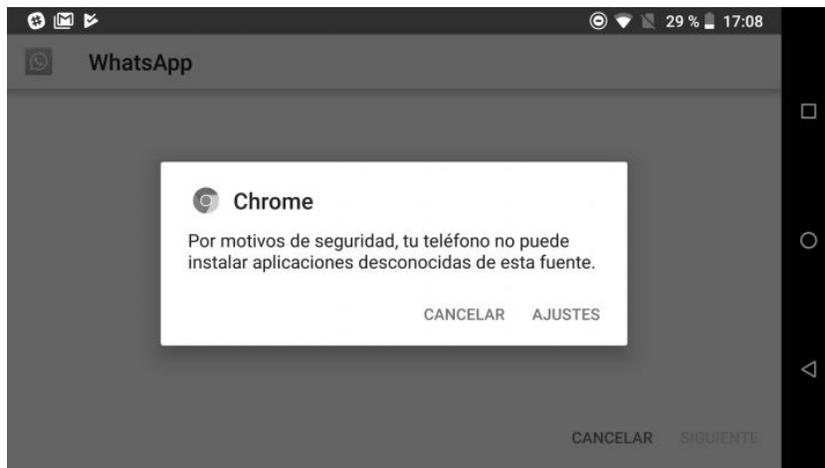


FIGURA 3.6 GUI - MENSAJE INFORMATIVO

- d) Dentro del mensaje de la Figura 3.6 existen dos opciones, se debe elegir la opción de “AJUSTES”, al seleccionar dicha opción esta te redirigirá a una nueva vista como la que se muestra en la Figura 3.7.

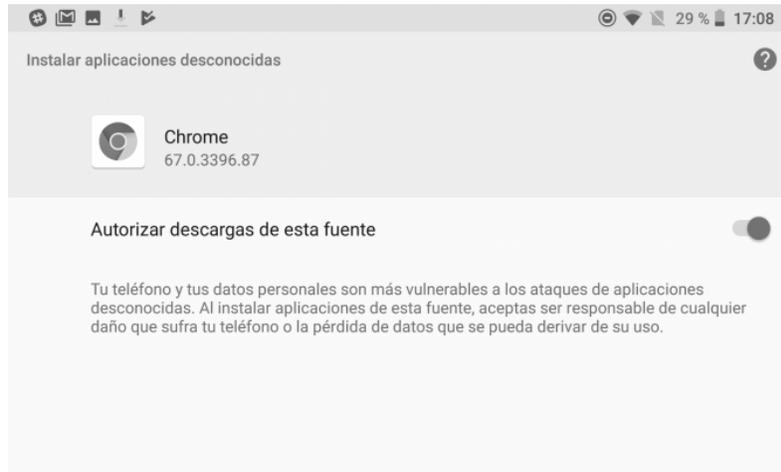


FIGURA 3.7 GUI - PERMISOS

- e) Deslizar la pestaña de la pestaña para activar el permiso “Autorizar descargas de esta fuente” de la Figura 3.7.
- f) Por último, se debe regresar a la instalación y concluir el proceso, en esta ocasión no muestra el mensaje de la Figura 3.6.

## Tarea 2: Configuración de la aplicación

La tarea de configuración de la aplicación contiene dos acciones principales que son: registro del usuario en la aplicación y registro del contacto de emergencia, configuración del mensaje y elección del contacto de emergencia, a continuación, se describe detalladamente como deben realizar estas acciones, para poder cumplir con esta tarea.

- 1) Registrar el usuario en la aplicación y registrar el contacto de emergencia
  - a. Para poder realizar esta acción el usuario debe entrar a la aplicación taras haber concluido el proceso de instalación, una vez que haya ingresado a la aplicación se muestra una interfaz idéntica a la de Figura 3.8.
  - b. Una vez ingresado a la pantalla se debe elegir el botón registro, para dar inicio al proceso de registro.
  - c. Al seleccionar el botón registro muestra la Figura 3.9, en la cual se solicitan los datos del usuario que son: nombre, edad, correo electrónico, género y teléfono. Adicionalmente solicita los datos de un primer contacto de emergencia los datos solicitados para este son: nombre, correo electrónico, teléfono.
  - d. Al ingresar todos los datos solicitados se debe oprimir el botón Guardar, el cual valida que los datos ingresados sean correctos y en caso de que algún dato no sea válido la aplicación indica el campo que es incorrecto.

- e. Una vez que la aplicación haya validado los datos, se registra el usuario y su contacto de emergencia en la base de datos.

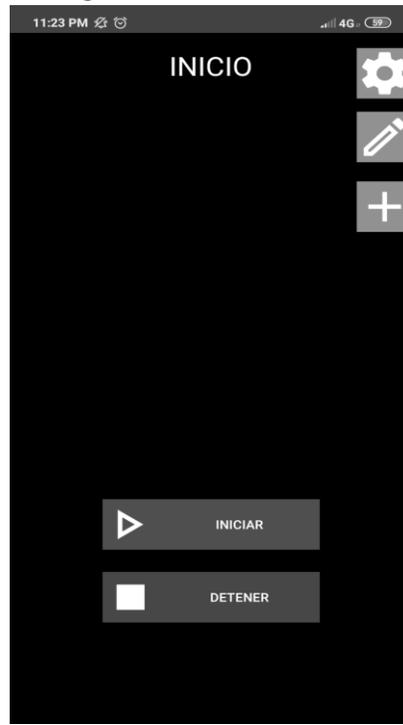


FIGURA 3.8 GUI - INICIO



FIGURA 3.9 GUI -REGISTRO DE USUARIO

- f. Tras haber concluido el registro del usuario la aplicación muestra un mensaje informativo y posteriormente se redirige a la interfaz principal de la aplicación, la cual se muestra en la Figura 3.10 y con esto concluye el proceso de registro del usuario en la aplicación y de registro del contacto de emergencia.



**FIGURA 3.10 GUI - PRINCIPAL**

- 2) Configuración del mensaje y elección del contacto de emergencia
- Para iniciar esta acción el usuario debe ingresar al módulo de configuración del mensaje, en la Figura 3.11 se muestra como ingresar a dicho módulo.
  - Al ingresar el módulo de configuración se muestra la Figura 3.12, la cual tiene un mensaje predefinido para la notificación y un ComboBox el cuál tiene el listado de contactos registrados en la aplicación.
  - Si se desea cambiar el mensaje el campo es completamente editable para que el usuario pueda personalizar el mensaje, adicionalmente se debe elegir un contacto del ComboBox para poder realizar el guardado.
  - Para guardar los cambios se debe oprimir el botón Guardar que se muestra en la Figura 3.12, tras oprimir dicho botón la aplicación valida la información ingresada y en caso de ser correcta guarda la configuración. Adicionalmente muestra un mensaje con el texto "operación realizada correctamente" y posteriormente regresa al menú principal.

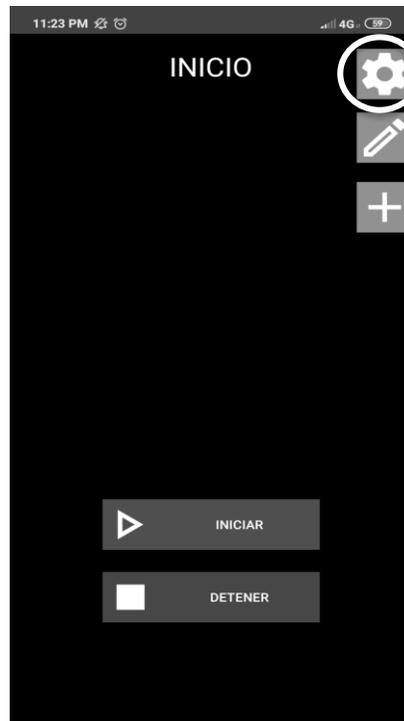


FIGURA 3.11 GUI - PRINCIPAL

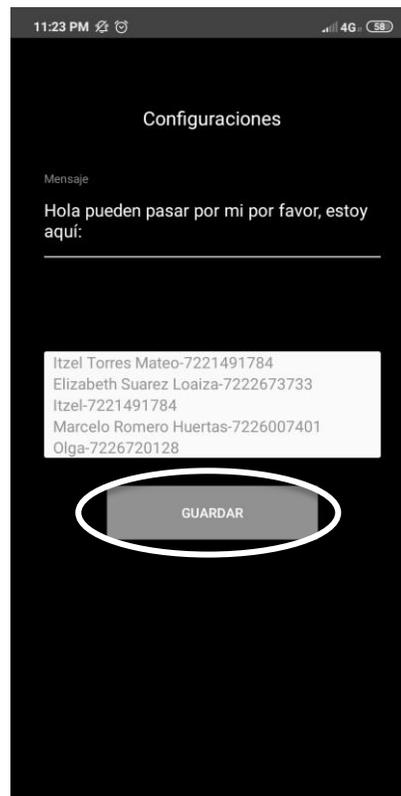


FIGURA 3.12 GUI - MÓDULO DE CONFIGURACIÓN DEL MENSAJE

### Tarea 3: Envío de alarma

La tarea de envío de alarma tiene dos acciones principales las cuales son: activar la interfaz gestual y realizar el movimiento para activar la interfaz gestual, a continuación, se describe detalladamente como se realizan las acciones, para poder cumplir con la tarea.

Para enviar la alarma se requiere haber concluido las dos tareas anteriores. La tarea instalación y configuración de la aplicación.

- 1) El usuario dentro de la aplicación debe ingresar a la interfaz principal de la aplicación que contiene un botón de iniciar el cual activa la interfaz gestual, dicho botón es señalado en la Figura 3.13.
  - a. Después de presionar el botón la interfaz gestual queda activa hasta que su desactivación con el botón detener que también se muestra en la Figura 3.13.

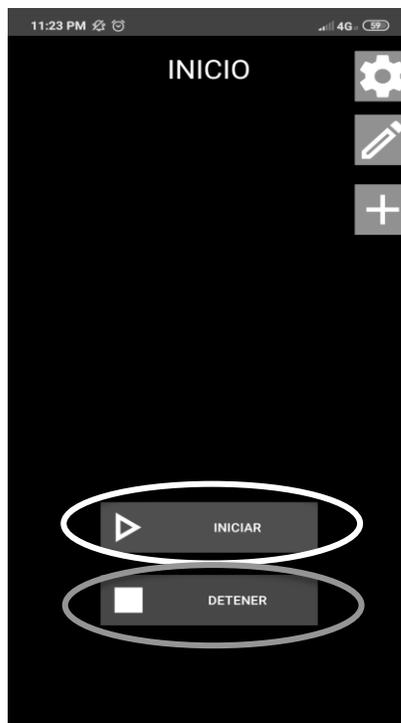
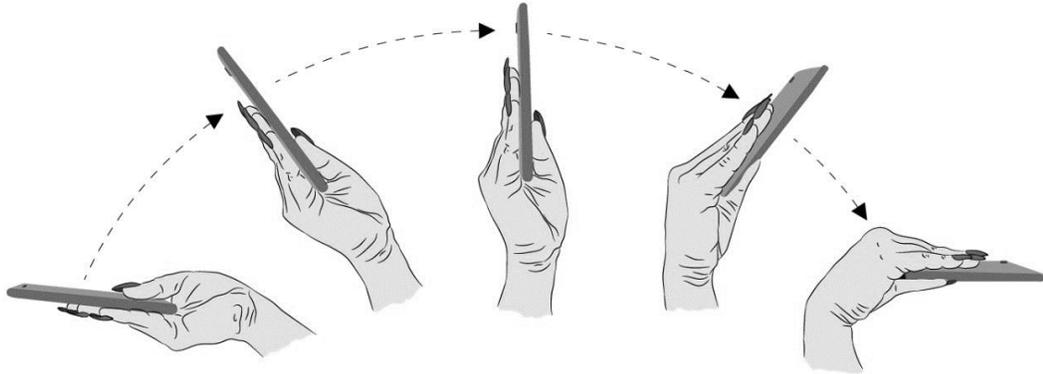


FIGURA 3.13 GUI - PRINCIPAL

- 2) Una vez activada la interfaz gestual, está siempre se encuentra monitoreando el estado del acelerómetro para poder leer los movimientos que se estén realizando con el teléfono inteligente, para comparar si el movimiento realizado es el propuesto en este trabajo. El cual es un movimiento de media luna de atrás hacia delante con la pantalla viendo hacia abajo, tal cual se muestra en la Figura 3.14.



**FIGURA 3.14 MOVIMIENTO A REALIZAR**

### 3.2.1.3 MÉTRICAS DE USABILIDAD

Empleando las métricas de usabilidad definidas en la Tabla 2.2, Sección 2.2 la evaluación para cada una de las métricas es:

- a. Número de tareas completadas exitosamente, se realiza un conteo del número de alarmas recibidas y no recibidas por el contacto de emergencia configurado. Esta medición nos servirá para comprobar la hipótesis planteada en esta tesis.
- b. Facilidad de uso de la aplicación, se toman los valores obtenidos en una escala de preferencias que se entrega al usuario tras haber concluido la prueba en dicha escala se encuentra en el Anexo A. Dentro de la escala se busca medir la facilidad de uso, apariencia de la interfaz y funcionalidad/eficiencia de la aplicación, esto nos ayuda cambiar aspectos específicos de la interfaz para que esta sea más amigable para el usuario.
- c. Tiempo que toma realizar las tareas asignadas, se recopilan los tiempos de cada una de las tareas definidas en la métrica, posteriormente se obtiene el tiempo medio que toma realizar cada una de las tareas y finalmente se realiza un análisis para ver que se tiene que cambiar para reducir el tiempo de alguna o todas las tareas definidas en la métrica.

### 3.2.1.4 ESCENARIO DE EXPERIMENTACIÓN

Como se refiere en la hipótesis postulada para esta tesis, se busca comprobar la exactitud de la aplicación para enviar una notificación verdadera-positiva. Por otro lado, como fue planteado en el objetivo general de esta tesis, la medición de la exactitud debe realizarse cuando el usuario se encuentra en situaciones de riesgo. Por esta razón, los escenarios planteados para esta tesis se consideran en la vía de tres colonias del Estado de México: Santa Cruz Atzacapotzaltongo y Nueva Oxtotitlán del municipio de Toluca, y El Calvario del municipio de Zinacantepec, para simular el escenario de robo a peatones. Esto se plantea de esta forma, ya que la mayoría de los usuarios que participaron en las pruebas tuvieron miedo de salir a los lugares plantados originalmente. Pero esta prueba se realiza empleado los datos móviles de sus teléfonos celulares.

### 3.2.1.5 PROCESO DE EVALUACIÓN

Para la evaluación de la interfaz gestual se toman las métricas de usabilidad definidas en la Tabla 2.2, Sección 2.2, ya que éstas son la base para medir la eficiencia, eficacia y el nivel de satisfacción de los usuarios que participaron en las pruebas realizadas.

En la Tabla 3.3 se observa la relación que existe entre las métricas de usabilidad definidas y las actividades realizadas por los usuarios en los escenarios planteados.

**TABLA 3.3 PROCESO DE EVALUCIÓN EN CUANTO A LAS MÉTRICAS**

Métrica	Actividades
<b>Facilidad de uso de la aplicación</b>	Para evaluar está métrica se toma en cuenta lo siguiente: <ul style="list-style-type: none"> <li>• El grado satisfacción que tienen los usuarios al estar utilizando la aplicación gestual</li> </ul>
<b>Tiempo que toma realizar las tareas asignadas</b>	Para medir está métrica se mide el tiempo que le toma a cada usuario realizar las actividades siguientes tareas. <ul style="list-style-type: none"> <li>• Registro de alarma</li> <li>• Envío de notificación</li> <li>• Instalación</li> <li>• Configuración</li> </ul>
<b>Número de operaciones exitosas</b>	Para medir esta métrica se realiza un conteo del número de mensajes recibidos una vez hecho el gesto que envía la notificación.

En las Tablas 3.4, 3.5, 3.6 y 3.7, se muestra la forma en la que se evalúa la métrica tiempo que toma realizar las tareas asignadas, dentro de la métrica se eligieron cuatro tareas a las cuales se les aplica la evaluación, ya que son las tareas primordiales dentro de la aplicación de interfaz gestual.

**TABLA 3.4 FORMA DE EVALUACIÓN DE MÉTRICA DE TIEMPO PARA LA CONFIGURACIÓN**

<b>Atributo</b>	Tiempo que toma realizar las tareas asignadas.
<b>Concepto de medición</b>	Se cronometra el tiempo que toma realizar la configuración de la aplicación.
<b>Método de medición</b>	Tiempo para configurar la aplicación de interfaz gestual.
<b>Nivel ahora</b>	45 segundos.
<b>Peor caso</b>	1 minuto con 30 segundos.
<b>Nivel planeado</b>	1 minutos.
<b>Mejor caso</b>	45 segundos.

TABLA 3.5 FORMA DE EVALUACIÓN DE MÉTRICA DE TIEMPO PARA LA INSTALACIÓN

<b>Atributo</b>	Tiempo que toma realizar las tareas asignadas.
<b>Concepto de medición</b>	Se cronometra el tiempo que toma realizar la instalación de la aplicación.
<b>Método de medición</b>	Tiempo para instalación la aplicación de interfaz gestual.
<b>Nivel ahora</b>	1 minuto.
<b>Peor caso</b>	2 minutos.
<b>Nivel planeado</b>	1 minuto.
<b>Mejor caso</b>	1 minuto con 30 segundos.

TABLA 3.6 FORMA DE EVALUACIÓN DE MÉTRICA DE TIEMPO PARA EL REGISTRO DE ALARMA

<b>Atributo</b>	Tiempo que toma realizar las tareas asignadas.
<b>Concepto de medición</b>	Se cronometra el tiempo que toma realizar el registro de alarma de la aplicación.
<b>Método de medición</b>	Tiempo que toma realizar el registro de alarma.
<b>Nivel ahora</b>	8 segundos.
<b>Peor caso</b>	16 segundos.
<b>Nivel planeado</b>	12 segundos
<b>Mejor caso</b>	8 segundos.

TABLA 3.7 FORMA DE EVALUACIÓN DE MÉTRICA DE TIEMPO PARA EL ENVÍO DE LA NOTIFICACIÓN

<b>Atributo</b>	Tiempo que toma realizar las tareas asignadas.
<b>Concepto de medición</b>	Se cronometra el tiempo que toma realizar el envío de la notificación de la aplicación.
<b>Método de medición</b>	Tiempo que toma realizar envío de la notificación.
<b>Nivel ahora</b>	8 segundos.
<b>Peor caso</b>	16 segundos.
<b>Nivel planeado</b>	12 segundos
<b>Mejor caso</b>	8 segundos.

En la Tabla 3.8 se muestra la forma de evaluación nos permite comprobar la hipótesis planteada.

**TABLA 3.8 FORMA DE EVALUACIÓN DE OPERACIONES EXITOSAS EN LA RECEPCIÓN DE NOTIFICACIONES**

<b>Atributo</b>	Número de operaciones exitosas
<b>Concepto de medición</b>	Se hace un conteo de notificaciones
<b>Método de medición</b>	Conteo de notificaciones recibidas por los contactos de emergencia
<b>Nivel ahora</b>	15 notificaciones
<b>Peor caso</b>	10 notificaciones
<b>Nivel planeado</b>	12 notificaciones
<b>Mejor caso</b>	15 notificaciones

En la Tabla 3.9 se muestra la forma de evaluación para medir el grado de satisfacción del usuario al utilizar la aplicación, esto con el objetivo de medir si la aplicación es visualmente atractiva y fácil de usar.

En la sección 3.2 se muestran los resultados obtenidos, en las pruebas con los usuarios.

### 3.3 RESULTADOS

En esta sección se muestra el resultado de evaluar las tareas con respecto a las métricas de usabilidad presentes en la Sección 2.2 en la Tabla 2.2.

#### 3.3.1 TAREAS Y RESULTADOS DE LA EVALUACIÓN

Para la métrica de tiempo se consideran cuatro actividades las cuales son: instalación, configuración, envío de notificación y registro de alarma. Se toma el tiempo que tarda cada usuario en realizar las tareas. Por lo que el tiempo medio que tarda cada usuario en completar cada tarea se encuentra en la Tabla 3.10.

**TABLA 3.9 FORMA DE EVALUACIÓN DE MÉTRICA DE FACILIDAD DE USO DE LA APLICACIÓN DE INTERFAZ GESTUAL**

<b>Atributo</b>	Facilidad de uso de la aplicación
<b>Concepto de medición</b>	Se busca medir la satisfacción de uso de aplicación
<b>Método de medición</b>	Medir la satisfacción de uso de aplicación con una encuesta de satisfacción
<b>Nivel ahora</b>	Excelente
<b>Peor caso</b>	Malo
<b>Nivel planeado</b>	Bueno
<b>Mejor caso</b>	Excelente

**TABLA 3.10 RESULTADOS DE LA MÉTRICA DE TIEMPO**

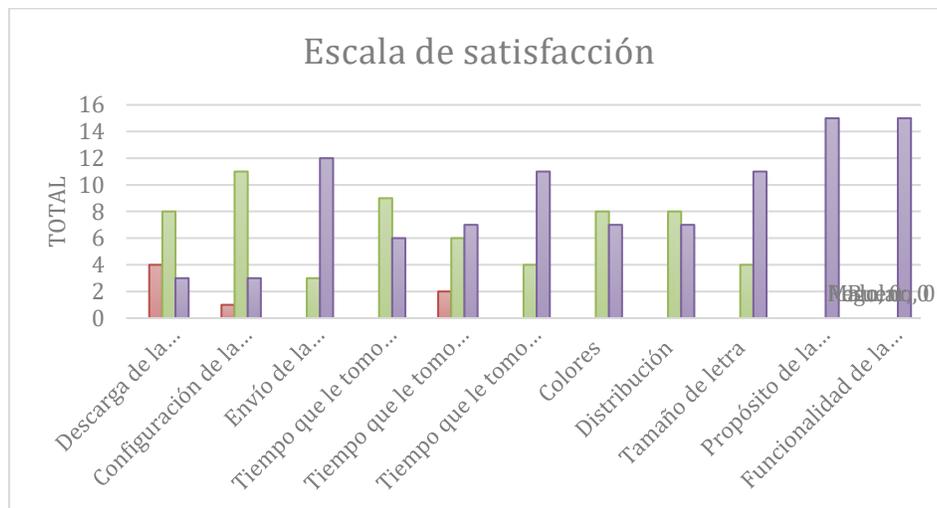
Criterio	Tiempo
<b>Instalación</b>	37.20 segundos
<b>Configuración</b>	48.43 segundos
<b>Registro de alarma</b>	12.00 segundos
<b>Envío de notificación</b>	12.00 segundos

En la Tabla 3.10, se encuentra el tiempo medio que tardan los usuarios en realizar las tareas asignadas y lo que se puede ver es que el tiempo de configuración es más alto a comparación de las demás tareas, porque dentro de la configuración se tiene la captura de la información del usuario, la captura de información del contacto de emergencia y la configuración del texto a enviar en la notificación. Mientras que el tiempo de instalación un poco elevado porque la descarga aplicación se hace desde *Google Drive*.

En cuanto a la métrica de facilidad de uso se emplea una encuesta de satisfacción donde se evalúan aspectos como: colores, tamaño, desempeño, etc., donde se le solicitó al usuario que indicará los aspectos que son malos, regulares, buenos y/o excelentes. En la Tabla 3.11 se muestra un resumen de los aspectos evaluados por el usuario.

En la Tabla 3.11 se contabilizan los criterios de la escala de satisfacción, se aprecia que la mayoría de los usuarios quedan satisfechos con la aplicación, sin embargo, algunas de las observaciones reportadas por los usuarios son: el poco agrado de la paleta colores elegida, el proceso de descarga no habitual y la distribución de los elementos de la *GUI* - Principal.

En la Gráfica 3.4 se presentan los aspectos mejor evaluados y los peor evaluados, de forma más detallada.



**GRÁFICA 3.3 GRÁFICA DE ESCALA DE SATISFACCIÓN**

**TABLA 3.11 RESUMEN DE LA EVALUACIÓN DE LA MÉTRICA DE FACILIDAD DE USO**

Criterio/Escala de satisfacción	Malo	Regular	Bueno	Excelente
Descarga de la aplicación	0	4	8	3
Configuración de la aplicación	0	1	11	3
Envío de la notificación de la aplicación	0	0	3	12
Tiempo que le tomo realizar el proceso descarga	0	0	9	6
Tiempo que le tomo realizar el proceso de configuración	0	2	6	7
Tiempo que le tomo realizar el envío de la notificación	0	0	4	11
Colores	0	0	8	7
Distribución	0	0	8	7
Tamaño de letra	0	0	4	11
Propósito de la aplicación	0	0	0	15
Funcionalidad de la aplicación	0	0	0	15

Por último, la métrica de operaciones realizadas correctamente, se cuantifica el número de mensajes recibidos y no recibidos por los contactos de emergencia. Donde se tiene que el 100% de los contactos elegidos recibieron los mensajes. Así como se muestra en la Tabla 3.12.

TABLA 3.12 RESULTADOS DE LA MÉTRICA DE NÚMERO DE OPERACIONES REALIZADAS CORRECTAMENTE

Criterio	Número
Notificaciones recibidas	15 mensajes
Notificaciones no recibidas	0 mensajes

Tal cual se muestra en la Tabla 3.12, todos contactos configurados como, contactos de emergencia recibieron la notificación con su respectiva ubicación.

En la Figura 3.15 se muestra un ejemplo del mensaje enviado durante las pruebas

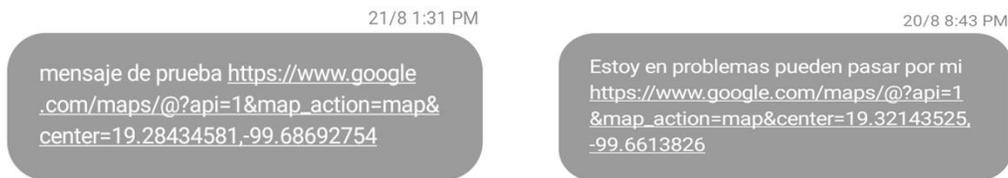


FIGURA 3.15 EJEMPLO DE MENSAJE ENVIADO DURANTE LAS PRUEBAS

En Figura 3.16 se muestra un ejemplo del mensaje que recibió uno de los contactos preferentes.



FIGURA 3.16 EJEMPLO DE MENSAJE RECIBIDO POR UN CONTACTO PREFERENTE DURANTE LAS PRUEBAS

En las Figura 3.17 se muestran fotos tomadas durante las pruebas.



FIGURA 3.17 EVIDENCIA DE LAS PRUEBAS

## CONCLUSIONES Y TRABAJO FUTURO

---

En esta tesis se analizó, diseñó y evaluó una aplicación móvil habilitada con interfaz gestual, la cual fue desarrollada para la plataforma operativa *Android* que permita leer y codificar un patrón de movimiento del brazo humano con la cual se envía una alarma, la cual puede ser útil cuando el usuario enfrenta situaciones de riesgo.

El motivo por el cual se decidió crear la aplicación de interfaz gestual en la plataforma propuesta, fue porque al contrario de otras plataformas operativas, *Android* no restringe el uso de los sensores del teléfono celular. Adicionalmente porque *Android* es de código libre y esto permite crear aplicaciones sin necesidad de pagar alguna licencia.

Este trabajo resume y aplica el enfoque innovador de desarrollo basado en metodologías y estándares, como el ISO 13407 del proceso de diseño centrado en el usuario. Con el fin de involucrar a los transeúntes en la mayor parte de las etapas de implementación, para lograr que estos puedan hacer uso de la aplicación y enviar la alarma con la mayor naturalidad posible.

Dentro de los resultados obtenidos en la evaluación de la aplicación móvil se distingue lo siguiente. Para la tarea *instalación de la aplicación* se obtiene un tiempo medio de 37.2 segundos. En cuanto a la tarea *configuración de la aplicación* se obtiene un tiempo medio de 48.43 segundos. Con respecto a la tarea *envío de la notificación* se obtiene un número de operaciones realizadas correctamente del 100% y con 12 segundos de tiempo medio. Finalmente, la tarea *registro de la alarma* se obtiene un tiempo medio de 12 segundos.

Como se muestra en la Tabla 3.12, Sección 3.2.1, es posible enviar al menos un 80% de alarmas verdaderas-positivas cuando el usuario ejecuta el movimiento de comando de la interfaz gestual, lo cual puede ser útil cuando el usuario enfrenta situaciones de peligro. Por lo tanto, se aprueba la hipótesis postulada en esta tesis y se cumple el objetivo general propuesto.

Como trabajo futuro se visualizan las siguientes líneas de investigación:

1. Realizar la lectura del gesto a través del giroscopio del teléfono inteligente en lugar de leer el movimiento por medio del acelerómetro, ya que los teléfonos celulares de las nuevas generaciones ya cuentan con dicho sensor. Adicionalmente, la lectura del gesto a través del giroscopio tiene mayor precisión para obtener los valores del movimiento comparado con las lecturas que se realizan con el acelerómetro.
2. Cambiar el campo de edad de la tabla víctima por fecha de nacimiento para mejorar el desempeño de la base de datos.
3. Crear un portal para visualizar en tiempo real las alarmas generadas por los incidentes, incluyendo la creación de un módulo de notificación a la autoridad correspondiente para atender el reporte.
4. Adicionalmente, los datos que se generen con el uso de la aplicación móvil propuesta en esta tesis, se pueden emplear para aplicar algún proceso de minería de datos que permita extraer información adicional asociada a este tipo de incidentes.
5. Finalmente, se debe aplicar la psicología de colores para mejorar la estética gráfica de *IGEA*.

---

## ANEXO A. ENCUESTA

Estimado usuario, el siguiente cuestionario pretende conocer su opinión con respecto a diversos aspectos que acaba de observar en la prueba de la aplicación, esto con el fin de mejorar el diseño y desempeño de la aplicación.

Por favor indica tu opinión con una X en cada una de las preguntas.

<b>Preguntas para medir la facilidad de uso de la interfaz</b>	Malo	Regular	Bueno	Excelente
La descarga de la aplicación fue:				
La configuración de la aplicación fue:				
El envío de la notificación de la aplicación fue:				
El tiempo que le tomo realizar el proceso descarga fue:				
El tiempo que le tomo realizar el proceso de configuración fue:				
El tiempo que le tomo realizar el envío de la notificación fue:				
<b>Preguntas para medir la apariencia de la interfaz</b>				
Los colores usados en la aplicación son:				
La distribución de elementos de la aplicación es:				
El tamaño de la letra usando en la aplicación es:				
<b>Preguntas para medir la funcionalidad</b>				
El propósito de la aplicación lo considera:				
En términos generales, la funcionalidad de la aplicación fue:				
Observaciones:				

---

## REFERENCIAS

1. Human Computer interaction. Alan Dix, Janet Finlay, Gregory D. Abowd, Russell Beale. Pearson Prentice Hall. Tercera Edición (2004)
2. User-Centered Website Development (a human-computer interaction approach). Daniel D. McCracken, Rosalee J. Wolfe. Pearson Prentice Hall. (2004)
3. Interaction Design: Beyond Human-Computer Interaction. Preece, Rogers and Sharp (Wiley & Sons, 2002).
4. Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests. Jeffrey Rubin, Dana Chisnell, Jared Spool. Wiley; 2 edition. ISBN-10: 0470185481
5. LookApp,  
[http://manzanamecanica.org/2014/09/prueba\\_esta\\_app\\_para\\_reportes\\_ciudadanos\\_sobre\\_asaltos\\_robos\\_y\\_acoso\\_callejero.html](http://manzanamecanica.org/2014/09/prueba_esta_app_para_reportes_ciudadanos_sobre_asaltos_robos_y_acoso_callejero.html)
6. HCI Beyond the GUI. Philip Kortum. The Morgan Kaufmann Series in Interactive Technologies. 2008. ISBN-10: 0123740177.
7. Ubiquitous Computing Smart Devices, Environments and Interactions. Stefan Poslad. ISBN 978-0-470-03560-3. First Edition 2009. John Wiley & Sons Ltd.
8. Developer.android.com. (2017). Android Developers. [online] Available at: <https://developer.android.com/index.html> [Accessed 5 Dec. 2016].
9. Lee, W. (2012). *Android 4: Desarrollo de aplicaciones*. Madrid: Anaya Multimedia.
10. Tasa de prevalencia delictiva INEGI (2017). <http://www.beta.inegi.org.mx/temas/victimizacion/>
11. R. Kwapisz Jennifer, M. Weiss Gary y A. Moore Samuel (2010) Activity Recognition using Cell Phone Accelerometers.
12. Pressman S. R., Ingeniería del software un enfoque práctico, tercera edición, Mc Graw-Hill, España, 1993.
13. J. Moran, "The command language grammar: a representation for the user interface o interactive systems," *International Journal of Man-Machine Studies*, 15(1) 1981.3-50.
14. Blake, J. (2012). The natural user interface revolution. En *Natural User Interfaces in .Net* (págs. 1-43). Manning.
15. Nielsen, M., Störing, M., Moeslund, T. B., & Granum, E. (2004). A procedure for developing intuitive and ergonomic gesture interfaces for HCI. *Gesture-Based Communication in Human-Computer Interaction*, 2915, 409-420.
16. Jégo, J. F., Paljic, A., & Fuchs, P. (2013). User-defined gestural interaction: A study on gesture memorization. *IEEE Symposium on 3D User Interfaces* (págs. 7-10). IEEE.
17. Nacenta, M. A., Kamber, Y., Qiang, Y., & Kristensson, P. O. (2013). Memorability of pre-designed and user-defined gesture sets. *Proc. SIGCHI Conference on Human Factors in Computing Systems* (págs. 1099-1108). ACM.
18. LaViola, J. J., & Keefe, D. F. (2011). 3D spatial interaction: applications for art, design, and science. *ACM Siggraph 2011 Courses*.
19. Vatavu, R. D. (2012). User-Defined Gestures for Free-Hand TV Control. *Proc. 10th European conference on Interactive tv and video* (págs. 45-48). ACM.
20. Walter, R., Bailly, G., Valkanova, N., & Müller, J. (2014). Cuenesics: Using Mid-Air Gestures to Select Items on Interactive Public Displays. *Proc. MobileHCI'14*. ACM.
21. Akazawa, N., Takei, Y., Nakayama, Y., Kakuda, H., & Suzuki, M. (2013). A learning support system for 9 × 9 multiplication table with Kinect. *Proc. IEEE 2nd Global Conference on Consumer Electronics* (págs. 253-257). IEEE.
22. Aumi, M. T., & Kratz, S. (2014). AirAuth: towards attack-resilient biometric authentication using in-air gestures. *CHI'14 Extended Abstracts on Human Factors in Computing Systems* (págs. 1585-1590). ACM.
23. Tuntakurn, A., Thongvigitmanee, S. S., Sa-Ing, V., Hasegawa, S., & Makhanov, S. S. (2013). Natural interactive 3D medical image viewer based on finger and arm gestures. *Proc. 6th Biomedical Engineering International Conference* (págs. 1-5). IEEE.

24. Garzotto, F., & Valoriani, M. (2013). Touchless gestural interaction with small displays: a case study. Proc. Biannual Conference of the Italian Chapter of SIGCHI. ACM.
25. Hayashi, E., Maas, M., & Hong, J. I. (2014). Wave to Me: User Identification Using Body Lengths and Natural Gestures. Proc. 32nd annual ACM conference on Human factors in computing systems (págs. 3453-3462). ACM.
26. Tan, J. H., Chao, C., Zawaideh, M., Roberts, A. C., & Kinney, T. B. (2013). Informatics in radiology: Developing a touchless user interface for intraoperative image control during interventional radiology procedures. *Radiographics*, 33(2), E61-E70.
27. Erazo, O., Pino, J. A., Pino, R., & Fernández, C. (2014). Magic Mirror for Neurorehabilitation of People with Upper Limb Dysfunction Using Kinect. En IEEE (Ed.), Proc. IEEE 47th Hawaii International Conference on System Sciences (págs. 2607-2615). IEEE.
28. Que es un Smartphone y Características. Teléfonos Inteligentes. (s.f.). areatecnologia. Recuperado 28 de septiembre de 2017, de <https://www.areatecnologia.com/Que-es-un-smartphone.htm>
29. Smartwatch Comparativas, Para qué sirven, Características, Precios, Análisis. (s.f.). areatecnologia. Recuperado 28 de septiembre de 2017, de <https://www.areatecnologia.com/smartwatch.html>
30. Phablet o Tablet Qué Son Ventajas Desventajas Precios. (s. f.). areatecnologia. Recuperado 28 de septiembre de 2017, de <https://www.areatecnologia.com/electronica/phablet.html>.
31. The University of California, Riverside, ed. (3 de diciembre de 2015). When Apps Talk Behind Your Back.
32. Fm, Y. (2019, 27 diciembre). 14 sensores que encontrarás en tu móvil: cómo funcionan y para qué sirven. Xataka. <https://www.xataka.com/basics/sensores-que-encontraras-tu-movil-como-funcionan-sirven>.
33. Arquitectura de la plataforma | Desarrolladores de Android. (s.f.). Android Developers. Recuperado 28 de septiembre de 2017, de <https://developer.android.com/guide/platform?hl=es-419>.
34. Arquitectura iOS - Tecnología iOS. (s. f.-b). sites.google. Recuperado 28 de septiembre de 2019, de <https://sites.google.com/site/tecnologiaiostm/desarrollo-de-aplicaciones/arquitectura-ios>.
35. Tipos de sistemas operativos móviles - Sistemas operativos. (s. f.). sites.google. Recuperado 28 de septiembre de 2017, de <https://sites.google.com/site/sistemasoperativosfranco9/tipos-de-sistemas-operativos-moviles>.
36. K.D Eason. Information Technology and Organizational Change. London: Taylor & Francis 1998.