

**UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA**



TESIS DE GRADO

**“APLICACIÓN MOVIL DE ASISTENCIA FAMILIAR INMEDIATA
PARA PERSONAS QUE PADECEN DE EPILEPSIA”**

PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA

MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: RONALD HERNAN CRUZ ARIAS

TUTOR METODOLOGICO: LIC. FREDDY MIGUEL TOLEDO PAZ

ASESOR: M.Sc. CARLOS MULLISACA CHOQUE

LA PAZ – BOLIVIA

2017



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



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria:

Dedicada a Dios y a la Virgen de Guadalupe por sus bendiciones...

A mis amados padres Wilma y Fructuoso que me dieron tanto, sin pedir nada a cambio...

A mis queridos hermanos Ruben, Lisseth, Carmen, Selene y Miguel por su apoyo incondicional...

A mi esposa querida Patty mi compañera de vida, la que supo apoyarme en mis aciertos y desaciertos...

A mis hijos Ruben y Luis la luz de mis ojos...

A mis compañeros de universidad que por diferentes razones no lograron culminar sus estudios...

A mi querido amigo de toda la vida Rodrigo Veliz Balboa...

AGRADECIMIENTOS

A tiempo de culminar esta tesis, deseo agradecer a nuestro Supremo Creador por haberme apoyado en cada instante de mi vida, permitiéndome concluir este ciclo y comenzar con uno nuevo...

A mi Docente Tutor: LIC. FREDDY MIGUEL TOLEDO PAZ, por permitir que lleve a cabo el presente trabajo, por los consejos, sugerencias y comprensión.

A mi Docente Asesor: LIC. CARLOS MULLISACA CHOQUE M.Sc, por los consejos, sugerencias y comprensión que me brindo a lo largo de la tesis.

A mis docentes Lic. Carmen Huanca y Lic. Javier Reyes por sus recomendaciones, por sus enseñanzas y por su apoyo incondicional.

A la Universidad Mayor De San Andrés por permitirme pertenecer a tan prestigiosa casa superior de estudios.

A mi bro Ruben Julio Cruz Arias por darme las fuerzas necesarias para seguir adelante, apoyándome en cada momento.

RESUMEN

La presente tesis de grado tiene como finalidad el desarrollo de una aplicación móvil georeferenciada de alerta a los familiares de la persona que sufre de convulsiones o ataques, con el fin de comunicar la necesidad de auxilio familiar por medio de alertas con el uso de móviles inteligentes con el apoyo de mapas de ubicación y así reducir el tiempo de asistencia médica en el momento en que se genera esta situación.

En el presente trabajo se utilizó la metodología Mobile – D que se destaca por conseguir ciclos de desarrollos muy rápidos en equipos muy pequeños. Según este método, trabajando de esa manera se deben conseguir productos totalmente funcionales en menos de diez semanas.

Se trata de un método basado en soluciones conocidas y consolidadas: Extreme Programming (XP), Crystal Methodologies y Rational Unified Process (RUP), XP para las prácticas de desarrollo, Crystal para escalar los métodos y RUP como base en el diseño de ciclo de vida

También se encuentran las métricas de calidad, para lo cual usamos parámetros de medición basados en la norma ISO 9126.

Para el análisis de costo usamos el modelo constructivo de Costo (o COCOMO, del inglés CONstructive COSt MOdel) es un modelo matemático de base empírica utilizado para estimación de costos para el desarrollo de un software.

Finalmente se presentan las conclusiones y recomendaciones.

Las conclusiones que reflejan los objetivos alcanzados y obtenidos con la presente tesis. Y las recomendaciones de la aplicación móvil para futuros proyectos

INDICE

1	MARCO INTRODUCTORIO	1
1.1	Antecedentes del trabajo	3
1.2	Planteamiento del problema	3
1.2.1	Problema general	3
1.2.2	Problemas específicos	4
1.3	Objetivos	4
1.3.1	Objetivo general.....	4
1.3.2	Objetivos específicos	5
1.4	Hipótesis	5
1.4.1	Variables de entorno	5
1.5	Justificación.....	5
1.5.1	Justificación técnica	5
1.5.2	Justificación económica	6
1.5.3	Justificación social	6
1.5.4	Justificación operativa.....	6
1.6	Alcances y Límites	7
1.6.1	Alcances	7
1.6.2	Límites.....	7
1.7	Aportes.....	7
1.7.1	Práctico.....	7
1.7.2	Teórico	8
1.8	Metodología	8
1.8.1	Tipo de investigación	8
1.8.2	Método aplicativo	8

2	MARCO TEÓRICO	10
2.1	Ingeniería de sistemas	11
2.1.1	Características de la ingeniería de sistemas	11
2.2	Ingeniería web	12
2.2.1	Proceso de la ingeniería web	12
2.3	Atributos aplicación web	13
2.4	Ingeniería móvil.....	14
2.4.1	Computación móvil.....	15
2.4.2	Dispositivo móvil.....	16
2.4.3	Sistema operativo móvil.....	17
2.5	Metodología de desarrollo Mobile D.....	18
2.5.1	Elementos	18
2.6	Geolocalización	20
2.7	API Google Maps	21
2.8	Modelo Vista Controlador.....	22
2.9	Calidad de software	23
2.10	Estimación de Costos COCOMO II.....	28
2.10.1	Estimación de esfuerzo.....	30
2.10.2	Modelo de composición de aplicación	30
3	MARCO PRÁCTICO.....	33
3.1	Introducción	33
3.2	Exploración	33
3.2.1	Recopilación de información	33
3.2.2	Identificación de actores.....	35

3.2.3	Requerimientos	36
3.3	Inicialización	41
3.3.1	Requerimientos del sistema	41
3.3.2	Requerimientos de la aplicación móvil	42
3.3.3	Requisitos del sistema	42
3.3.4	Sistema propuesto	43
3.3.5	Modelado de procesos.....	44
3.4	Productización.....	45
3.4.1	Planificación	45
3.4.2	Cronograma de trabajo	46
3.4.3	Diseño de la base de datos.....	47
3.4.4	Diseño físico de la base de datos	48
3.4.5	Diagrama de clases.....	50
3.4.6	Diseño de pantallas web.....	51
3.4.7	Diseño de pantallas aplicación móvil.....	54
3.5	Desarrollo del sistema.....	56
3.5.1	Sistema web.....	57
3.5.2	Sistema móvil	61
3.6	Estabilización	64
3.7	Pruebas.....	64
3.7.1	Resultados de la prueba.....	66
3.8	Calidad de Software.....	67
3.8.1	Funcionalidad	67
3.8.2	Fiabilidad.....	70

3.8.3	Facilidad de mantenimiento.....	71
3.8.4	Eficiencia.....	71
3.8.5	Flexibilidad.....	73
3.9	Estimación de costos.....	73
4	CONCLUSIONES Y RECOMENDACIONES.....	78
4.1	Conclusiones	78
4.1.1	De la hipótesis.....	78
4.1.2	De los objetivos	78
4.2	Recomendaciones	79
5	Bibliografía.....	80

INDICE DE TABLAS

Tabla 2.1 Diferencia entre aplicación móvil y de escritorio.....	14
Tabla 2.2 Métricas de adecuadad	25
Tabla 2.3 Métrica de entendibilidad.....	26
Tabla 2.4 Métrica de portabilidad.....	27
Tabla 2.5 Distribución del mercado de software	28
Tabla 3.1 Resultados de encuesta	35
Tabla 3.2 Historia de usuario 1.....	37
Tabla 3.3 Prueba de aceptación: Información	37
Tabla 3.4 Historia de usuario 2.....	38
Tabla 3.5 Prueba de aceptación Registro de usuarios	38
Tabla 3.6 Historia de usuario 3.....	38
Tabla 3.7 Prueba de aceptación: Alerta.....	39
Tabla 3.8 Historia de usuario 4.....	39
Tabla 3.9 Prueba de aceptación: Video.....	39
Tabla 3.10 Historia de usuario 5.....	40
Tabla 3.11 Prueba de aceptación: Estadísticas	40
Tabla 3.12 Historia de usuario 6.....	40
Tabla 3.13 Consultas.....	41
Tabla 3.14 Tabla de Requerimientos	41
Tabla 3.15 Tabla de requerimientos de la aplicación móvil.....	42
Tabla 3.16 Planificación de tareas	46
Tabla 3.17 Tareas y su duración	46

Tabla 3.18 Estructura de la tabla usuario	48
Tabla 3.19 Estructura de la tabla tUsuario	49
Tabla 3.20 Estructura de la tabla tContactanos.....	49
Tabla 3.21 Estructura de la tabla historial	50
Tabla 3.22 Resultados de la prueba	66
Tabla 3.23 Resultados finales de evaluación.....	67
Tabla 3.24 Valores de Ajuste de complejidad	67
Tabla 3.25 Ajuste de complejidad del Punto Función.....	68
Tabla 3.26 Funcionalidad del sistema.....	69
Tabla 3.27 Resultados de evaluación de ejecución del sistema.....	70
Tabla 3.28 Resultados de evaluación de ejecución del sistema.....	71
Tabla 3.29 Resultados de evaluación según encuesta a los usuarios	72
Tabla 3.30 Resultados de evaluación de eficiencia del sistema.....	72
Tabla 3.31 Costo estimado de desarrollo de software.....	76

INDICE DE FIGURAS

Figura 2.1 Ingeniería de sistemas como puente	11
Figura 2.2 Procesos no soportados por la computación móvil	15
Figura 2.3 Paradigma de computación móvil	16
Figura 2.4 Formularios ubicados en la aplicación central	17
Figura 2.5 Ciclo de desarrollo Mobile-D	18
Figura 2.6 Modelo vista controlador	23
Figura 2.7 Factores de evaluación de calidad	24
Figura 3.1 Fases de Mobile-D	33
Figura 3.2 Arquitectura del sistema Epimovil	43
Figura 3.3 Arquitectura del sub-sistema móvil.....	44
Figura 3.4 Diagrama de contexto del sistema.....	44
Figura 3.5 Caso de uso general del sistema.....	45
Figura 3.6 Desarrollo de tareas	47
Figura 3.7 Tablas de la Base de datos	48
Figura 3.8 Diagrama de clases.....	50
Figura 3.9 Pantalla de ingreso web	51
Figura 3.10 Pantalla de ayuda.....	52
Figura 3.11 Pantalla de ingreso web	52
Figura 3.12 Pantalla de Administración web.....	53
Figura 3.13 Pantalla que muestra los lugares geográficos.....	53
Figura 3.14 Pantalla CRUD de administradores.....	54
Figura 3.15 Pantalla de ingreso móvil.....	55
Figura 3.16 Pantalla móvil de registro de usuario	55

Figura 3.17 Pantalla de ingreso móvil.....	56
Figura 3.18 Pantalla de ingreso móvil.....	56
Figura 3.19 Pantalla de envío de solicitud de ayuda	57
Figura 3.20 Pantalla de envío de solicitud de ayuda	57
Figura 3.21 Pantalla de envío de solicitud de ayuda	58
Figura 3.22 Pantalla de envío de solicitud de ayuda	59
Figura 3.23 Pantalla de envío de solicitud de ayuda	59
Figura 3.24 Pantalla de envío de solicitud de ayuda	60
Figura 3.25 Pantalla de envío de solicitud de ayuda	61
Figura 3.26 Pantalla de envío de solicitud de ayuda	62
Figura 3.27 Pantalla de envío de solicitud de ayuda	62
Figura 3.28 Pantalla de envío de solicitud de ayuda	63
Figura 3.29 Pantalla de envío de solicitud de ayuda	63
Figura 3.30 Resultados de la evaluación.	66
Figura 3.31 Datos de entrada COCOMO II.....	74
Figura 3.32 Datos de salida COCOMO II	75



CAPITULO I

MARCO INTRODUCTORIO

1 MARCO INTRODUCTORIO

Introducción

La epilepsia¹ es una enfermedad provocada por un desequilibrio en la actividad eléctrica de las neuronas de alguna zona del cerebro. Se caracteriza por uno o varios trastornos neurológicos que dejan una predisposición en el cerebro a padecer convulsiones recurrentes, que suelen dar lugar a consecuencias neurobiológicas, cognitivas y psicológicas. Se presenta cuando los cambios permanentes en el tejido cerebral hacen que el cerebro esté demasiado excitable o irritable y como resultado de esto, el cerebro envía señales anormales, lo que ocasiona convulsiones repetitivas e impredecibles.

De acuerdo con datos estadísticos obtenidos del Programa de Registro Único de Personas con Discapacidad, de 10.227.229 de habitantes en el país, 83. 192 presentan algún tipo de capacidad diferente, de los cuales 40. 368 fueron calificados, y de esta cifra, 940 enfrentan la enfermedad de la epilepsia, lo que representa que algo menos del 3% de la población sufre este trastorno.[1]

Android es un sistema operativo basado en Linux diseñado originalmente para dispositivos móviles, tales como teléfonos inteligentes, tablets, pero que actualmente se encuentra en desarrollo para usarse en netbooks y PCs. Fue desarrollado inicialmente por Android Inc., una firma comprada por Google en 2005. Es el principal producto de la Open Handset Alliance, un conglomerado de fabricantes y desarrolladores de hardware, software y operadores de servicio. Por otra parte, un 83% de los bolivianos posee un teléfono celular, según se desprende de una encuesta realizada durante el mes de septiembre en La Paz, Cochabamba, Santa Cruz y El Alto.[2]

Según la Universidad Politécnica de Valencia, define la georreferenciación, como un proceso por el cual se dota de un sistema de referencia de coordenadas terreno a una imagen digital que originariamente se encuentra en coordenadas pixel.

Por su parte, la geolocalización, atendiendo a la definición que ofrece Techopedia, geolocalización se define como la identificación de la ubicación de un dispositivo por ejemplo

¹Del latín epilepsia, a su vez del griego ἐπιληψία, 'intercepción'.

un radar, teléfono móvil o cualquier aparato tecnológico conectado a internet. Está relacionada con los sistemas de detección de posición, pero añade datos como información de la zona, calles, locales, etc.

Entonces, con el uso de la tecnología descrita, se plantea el desarrollo de un aplicativo móvil que permita la asistencia familiar inmediata para cualquier persona que presente convulsiones, para que esta pueda comunicar a los familiares de tal situación, incluyendo la ubicación donde se encuentra la persona.

1.1 Antecedentes del trabajo

- “Herramienta Metodológica para el desarrollo de aplicaciones web móvil” de Salgueiro, en cuyo trabajo propone una metodología para el desarrollo de una aplicación web móvil acorde a las necesidades de los usuarios desde el instante en que se inicia la planificación hasta la conclusión final.
- Sistema de administración de Historias clínicas: Caso Clínica Sanjinés” (2009) elaborado por Américo Guillermo Machicao Rejas, el cual automatizo el registro de historias clínicas de forma electrónica, la consulta de historias clínicas, además de realizar el seguimiento y control de pacientes y asignar una consulta, con metodología UML.
- “Sistema de ubicación o localización móvil basado en dispositivos móviles” propuesto por Guarachi, propone un sistema de ubicación capaz de explotar los servicios de localización combinados con las tecnologías usadas para el desarrollo de las aplicaciones móviles, tecnologías y servicios web.

1.2 Planteamiento del problema

1.2.1 Problema general

El estado epiléptico en el que se encuentra una persona representa una urgencia médica que debe ser atendida porque se tienen convulsiones acompañadas de intensas contracciones musculares, no puede respirar adecuadamente y presenta extensas descargas eléctricas en el cerebro. Si no se procede al tratamiento inmediato, el corazón y el cerebro pueden resultar permanentemente lesionados y puede sobrevenir la muerte.[3]

“De las 40.368 personas calificadas con discapacidad e identificadas en el país, cuantifica que 940 personas tienen epilepsia, lo que equivale a un 2,33%. Asimismo, la población masculina es identificada con mayores casos aproximadamente 509 personas y 431 en el caso de las mujeres”[1].

En el medio social boliviano, actualmente no se dispone de una herramienta software que sea utilizado por las personas que sufren convulsiones para informar al entorno familiar y recibir asistencia médica inmediata, lo que afecta a las personas.

Entonces, el problema de investigación es:

¿Una aplicación móvil de alerta familiar inmediata para personas que padecen de epilepsia, logra disminuir los riesgos que corren los afectados por este trastorno cerebral y ser asistidos de manera oportuna en el momento que se genere tal situación?

1.2.2 Problemas específicos

Los problemas que actualmente se presentan son:

- Los procedimientos que se realizan al momento en que se presentan ataques de epilepsia en las personas así como los familiares que pueden brindar ayuda son poco efectivos dado que se desconoce el momento y la ubicación geográfica del paciente.
- No se dispone de un medio de comunicación cuando la persona sufre una convulsión o ataque de epilepsia.
- El control y seguimiento de la enfermedad es realizado de manera irregular y con permanentes errores en el historial clínico de la persona con epilepsia.

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar una aplicación móvil georeferenciada de alerta familiar destinada a auxiliar de manera oportuna a las personas que padecen de epilepsia por medio de alertas vía telefonía a los familiares y personas cercanas del afectado incluyendo un mapa de ubicación y así reducir el tiempo de asistencia médica en el momento en que se genera esta situación.

1.3.2 Objetivos específicos

- Diseñar el sistema web-móvil que coadyuve a la asistencia médica basado en una metodología de desarrollo de software de programación ágil que permita utilizar los dispositivos inteligentes en la solicitud de ayuda inmediata.
- Establecer un medio de alerta y comunicación telefónica dirigido al grupo familiar cuando una persona sufre una convulsión o ataque de epilepsia para pedir ayuda médica.
- Registrar la frecuencia y lugar donde la persona sufre estas convulsiones de epilepsia para ser almacenados en un servidor de base de datos como archivo histórico.

1.4 Hipótesis

Una aplicación móvil georeferenciada permite coadyuvar en el auxilio inmediato a personas que padecen de epilepsia de forma que se refleje en un mapa la ubicación exacta de la persona reduciendo así el tiempo de asistencia.

1.4.1 Variables de entorno

Variable Independiente

X: Aplicación móvil

Variable Dependiente

Y: Tiempo de ayuda necesaria para el auxilio de la persona que sufre convulsiones.

Variable moderante

Entorno familiar, Centros médicos de salud

1.5 Justificación

1.5.1 Justificación técnica

Se propone implementar esta aplicación móvil que estará disponible solo bajo plataforma Android y dado que gran parte de la población dispone de celulares Android, su aplicación se justifica técnicamente.

Adicionalmente, este trabajo se justifica por que la estructura Cliente-Servidor basada en las tecnologías de hardware, software, bases de datos y redes, lo que permitirá una mayor interactividad del flujo de información y usuarios. El presente trabajo colaborará de alguna manera a la investigación de la forma en cómo se realizan aplicaciones móviles utilizando herramientas de desarrollo de aplicaciones móviles y web.

1.5.2 Justificación económica

Muchas personas que sufren de esta enfermedad, y cuando se presenta una convulsión, generalmente no son auxiliadas y son objeto de hurtos y robos.

Por tanto, desarrollar esta herramienta que permita a la persona comunicar y solicitar ayuda reduce los riesgos de acciones delincuenciales a la que está expuesto la persona. Entonces, se justifica económicamente ya que con esta aplicación Android se reducirán los costos de registro y transferencia de información desde el dispositivo móvil hacia la aplicación web encargada de centralizar toda esta información, enviando un mensaje de alerta al entorno familiar.

1.5.3 Justificación social

Si bien la aplicación móvil de alerta permite comunicar una situación de emergencia al entorno familiar, es posible establecer convenios con servicios de radio taxi por ejemplo o bien con una clínica que disponga de ambulancias y que ante una situación de emergencia, de manera inmediata acuda a ayudar a la persona hasta el punto donde se encuentra.

Por otra parte, cualquier Centro Médico podrá ofertar a un grupo social un medio de seguimiento de pacientes que sufren de ataques de epilepsia.

Finalmente, esta aplicación ofrecerá a los familiares un medio de ayuda inmediata además de permitir el seguimiento de la persona con convulsiones en el momento que se presente.

1.5.4 Justificación operativa

El sistema móvil de alerta para personas que sufren de epilepsia estará conformado por dos módulos:

- Un aplicativo APP, destinado a instalarse en los dispositivos inteligentes donde la persona debe registrarse y desde ese instante debe estar en permanente ejecución, y

cuando se presente un ataque de epilepsia, el usuario podrá mediante un botón siempre visible en el celular, enviar mensajes de alerta al entorno familiar y servicios de emergencia (servicio de transporte de radiotaxis y centros médicos) solicitando ayuda inmediata,

- Una aplicación web, donde se realizará el monitoreo de las personas registradas y generar reportes y consultas del grupo registrado.

1.6 Alcances y Límites

1.6.1 Alcances

Entre los alcances del presente trabajo se pretende:

- ⇒ Realizar el desarrollo de un prototipo de aplicación para dispositivos móviles que funcionan bajo plataforma Android que permita mandar mensajes de alerta al círculo de ayuda.
- ⇒ Mediante la plataforma web, se centralizará la información de las personas que serán sometidas a pruebas.
- ⇒ Para la validación de los resultados, se establecerán pruebas de aceptación mediante encuestas ejercidas sobre una muestra de personas establecidas mediante técnicas estadísticas y determinar la valoración de la aplicación móvil.

1.6.2 Límites

- ⇒ El presente proyecto trabajará en el envío de mensajes de alerta a un grupo familiar y servicios de emergencia a Centros de Salud.
- ⇒ La plataforma de trabajo será exclusivamente bajo Windows y la aplicación móvil trabajará bajo entorno Android.
- ⇒ No podrá ser aplicada a otras especialidades por la naturaleza propia de la enfermedad.

1.7 Aportes

1.7.1 Práctico

El principal aporte a la sociedad es que se presenta una herramienta de comunicación de la persona que sufre ataques de epilepsia hacia los familiares cercanos, solicitando ayuda inmediata.

Por otra parte, la aplicación misma registrará el instante y frecuencia con que se presentan estos ataques convulsivos, presentado así al médico especialista de un registro médico sobre el cual pueda establecer un diagnóstico.

1.7.2 Teórico

Ampliar las técnicas de desarrollo de aplicaciones móviles utilizando metodologías apropiadas para su elaboración.

1.8 Metodología

1.8.1 Tipo de investigación

Este trabajo se basará principalmente en el método científico cuyas etapas principales son:

Observación

Formulación de hipótesis

Experimentación

Emisión de conclusiones

1.8.2 Método aplicativo

- MOBILE-D

Se podría pensar que Mobile-D es una creación un tanto antigua, ya que se desarrolló como parte de un proyecto finlandés, ICAROS, en 2004.

El ciclo del proyecto se divide en cinco fases: exploración, inicialización, productización, estabilización y prueba del sistema. En general, todas las fases (con la excepción de la primera fase exploratoria) contienen tres días de desarrollo distintos: planificación, trabajo y liberación. Se añadirán días para acciones adicionales en casos particulares

- UML

Para el desarrollo de cualquier aplicación de software, es necesario seguir una metodología de la Ingeniería web principalmente para tener una orientación sobre los procesos que se deben seguir en todo momento y llegar a la conclusión con un trabajo perfectamente documentado.

- Técnicas

La entrevista personal: Consiste en un intercambio directo de información entre el Analista y un integrante de la organización. Se tiene una gran flexibilidad en la búsqueda de datos y brinda la oportunidad de entrar en contacto directo con el personal.

La encuesta escrita: Se desarrolla mediante el diseño de cuestionarios específicos que se dirigen a los empleados de la empresa vinculados con la investigación.

El estudio de documentación: Involucra la búsqueda y análisis de documentos existentes vinculados al estudio.

- Instrumentos.

Se utilizará herramientas CASE para el diseño del aplicativo, como el StarUML, para el modelado del sistema y la generación de las plantillas básicas de programación.

Para elaborar el código del programa, se utilizará el lenguaje PHP que ofrece diversas características de desarrollo y uso, pero lo más importante es que es de uso libre y por tanto no requiere licencias de uso. También se utilizarán otras herramientas como HTML5, BOOTSTRAP, CSS3 y archivos XML.

Para el almacenamiento, diseño y construcción de la base de datos, se hará uso del gestor de base de datos MySQL.

Para la aplicación móvil se hará uso de Android Studio, que es una plataforma libre para el desarrollo de aplicaciones móviles en plataforma Android.

Finalmente, para la representación de los mapas georreferenciados, se hará uso de Google Maps, que es un servidor de aplicaciones de mapas en la web.



CAPITULO II MARCO TEORICO

2 MARCO TEÓRICO

2.1 Ingeniería de sistemas

La ingeniería de sistemas puede definirse como la aplicación de técnicas científicas para transformar una necesidad operativa en la descripción de los parámetros de prestaciones de un sistema y en su configuración mediante la utilización de un proceso iterativo de definición, síntesis, análisis, diseño, prueba y evaluación; integrar los parámetros técnicos relacionados y asegurar la compatibilidad de todas las interrelaciones físicas, funcionales y del programa de forma que se consiga la mejor definición y diseño del sistema completo, como se muestra en la figura 2.1.[4]

Además de integrar los aspectos de fiabilidad, mantenibilidad, seguridad, supervivencia, de personal y otros similares en el proceso global de ingeniería para conseguir los objetivos técnicos, de coste y de calendario fijados. Incluye la integración oportuna de los factores técnicos, las relaciones funcionales y las actividades del programa. Esta incluye las diferentes disciplinas de diseño que se combinan e integran para conseguir el desarrollo de la obtención de un sistema que cubra las necesidades del consumidor de forma efectiva y eficaz. [4]

Figura 2.1 Ingeniería de sistemas como puente



Fuente: [4]

2.1.1 Características de la ingeniería de sistemas

La aplicación de los principios de la ingeniería de sistemas constituye más bien un proceso intelectual o una forma de organizar trabajos. Requiere un cambio de mentalidad para muchos, o un cambio de cultura.[4]

La ingeniería de sistemas pone un énfasis especial en determinadas áreas, y cabe señalar que es necesario:

- a) Utilizar un enfoque de arriba-abajo viendo al sistema como un todo. Aunque los trabajos de ingeniería del pasado lograron diseños muy satisfactorios de los diferentes componentes de un sistema, carecían de visión global y comprensión de cómo debían integrarse eficazmente todos ellos entre sí.[4]
- b) Contemplar todo el ciclo de vida del sistema y todas sus fases, que incluye el diseño y desarrollo del sistema, la producción y/o construcción, su distribución, su vida operativa, el apoyo y mantenimiento durante la misma, su baja y retirada (desecho). En el pasado la mayor atención se centraba solo en las actividades del diseño o adquisición del sistema, sin estar al tanto del impacto que las mismas podrían provocar en los aspectos de producción, vida operativa, y apoyo logístico para poder evaluar adecuadamente los riesgos asociados con las decisiones adoptadas en el proceso inicial de toma de decisiones, es necesario que las mismas se basen en consideraciones del ciclo de vida. [4]

2.2 Ingeniería web

La ingeniería web como la aplicación de principios científicos de ingeniería y administración de forma sistemática y disciplinada para crear, implantar y mantener aplicaciones y sistemas web de alta calidad. [5]

Pressman sostiene que la ingeniería web, al igual que la ingeniería de software, dicta un enfoque disciplinado para el desarrollo de un sistema basado en computadora.[5]

Por tanto, la ingeniería web puede definirse como el desarrollo de aplicaciones de tipo web, de manera ordenada, metódica y disciplinada; empleando las prácticas de ingeniería de sistemas e ingeniería del software para garantizar el funcionamiento del producto y facilitar su administración. Su objetivo principal es brindar soluciones y optimizar los problemas que afectan directamente a la humanidad.

2.2.1 Proceso de la ingeniería web

El desarrollo de aplicaciones web de alta calidad es la elaboración de un plan sistemático tanto para el diseño como para la implementación, al igual que el diseño de arquitectura, pruebas, evaluación y la incorporación de medidas de seguridad. Estos sistemas deben ser entendidos desde varios niveles de abstracción y examinados desde muchas perspectivas. De acuerdo a las características de estas aplicaciones, Pressman propone un modelo de proceso iterativo e

incremental, de carácter general, utilizado en el desarrollo de la mayoría de sistemas basados en web. [5]

- **Formulación:** Fase inicial del proceso de ingeniería web, en la que se establecen los objetivos de aplicación.
- **Planificación:** Estimación de costos globales de desarrollo y proyección detallada del tiempo de desarrollo.
- **Análisis:** Especifica los requisitos técnicos para la aplicación web, los elementos de contenido que serán incorporados y los requisitos de diseño gráfico (estética).
- **Ingeniería:** Comprende dos tareas paralelas: la primera consiste en el diseño de contenidos y producción, reúne todo el contenido de texto y multimedia que se vaya a integrar en, la aplicación web, y la segunda se centra en el diseño arquitectónico, de navegación y de interfaz.
- **Generación** de prototipos y pruebas: Etapa de construcción, en la que se fusionan las dos tareas de la actividad de ingeniería para elaborar páginas web ejecutables, de las cuales se prueban la navegabilidad, la funcionalidad y la exactitud de sus procesos.
- **Evaluación por el cliente:** Evalúa cada incremento del proceso de ingeniería web y los resultados que produce se convierten en la formulación del siguiente incremento.

Luego de analizar el proceso de ingeniería web, se revisan las características comunes de las aplicaciones web.

2.3 Atributos aplicación web

Los atributos comunes en toda aplicación web son: [5]

- Los atributos de Red: por su naturaleza, las aplicaciones web residen en una red y brindan servicios a las necesidades de diversos clientes. Pueden residir en internet, permitiendo el acceso desde cualquier lugar del mundo, en una intranet a través de la comunidad de redes de una organización, o en una extranet mediante comunicación entre redes.
- Controlada por el contenido, la función principal de las aplicaciones web es el empleo de hipertexto para presentar el contenido al usuario.

- Evolución continua, a diferencia de los sistemas de escritorio, cuya evolución es cronológicamente planificada en versiones específicas, las aplicaciones web están en constante cambio, no es de extrañar la actualización de su contenido cada hora o cada minuto.
- Inmediatez, el tiempo que se tarda en desarrollar una aplicación basada en web puede reducirse a días o semanas.
- Seguridad, las aplicaciones web requieren fuertes medidas de seguridad en toda su infraestructura, así como dentro de la misma aplicación.
- Estética, características fundamentales de las aplicaciones basadas en web son su apariencia y su interacción, tienen una gran influencia al momento de vender productos o ideas.

También se puede hacer una comparación o diferenciación entre una aplicación web y una aplicación de escritorio como se observa en la tabla 2.1.

APLICACIÓN WEB	APLICACIÓN DE ESCRITORIO
<ul style="list-style-type: none"> • Se puede acceder a través de un navegador. • La información que contiene por lo general es estática. • Al ser aplicación web es portable y se puede acceder a través de cualquier dispositivo siempre y cuando tenga un browser. • La seguridad depende del desarrollador, pero pueden ser muy seguras. • Requiere de conexión a internet en algunos casos. 	<ul style="list-style-type: none"> • Se requiere la instalación de un programa para acceder. • La información se puede ser estática o dinámica. • Solo se puede acceder a través de máquinas que tengan instalado la aplicación. • En cuanto a seguridad se considera que son muy seguras dependiendo del desarrollador de la aplicación. • La conexión a internet no es muy requerida,

Tabla 2.1 Diferencia entre aplicación móvil y de escritorio
Fuente:[5]

2.4 Ingeniería móvil

El sistema operativo que posee esta clase de terminales ofrece un sinnúmero de posibilidades, como si de un ordenador se tratará; y todo esto combinado con la actual potencia de procesamiento y capacidad de almacenamiento que permiten la ejecución no sólo de aplicaciones ligeras sino de

una gran parte de las pesadas; sin olvidar otras funcionalidades incorporadas a los móviles, como son el GPS, pantallas táctiles, llamadas telefónicas, posibilidad de enviar mensajes cortos.[6]

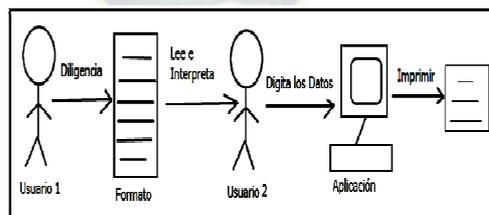
2.4.1 Computación móvil

Anteriormente muchas aplicaciones o programas se diseñaban y construían con interfaces exclusivas para entrada de datos provenientes de un teclado de computador. Además, se elaboraba un formato o formulario para llenarse a mano en el lugar donde sucedía la acción o se establecía la fuente de datos; los datos se ingresaban en la aplicación a partir de la información contenida en los formatos. [6]

El formato diligenciado, a mano, se entregaba a una persona; esta persona interpretaba los datos escritos, después digitaba lo interpretado, de la información contenida en el formato, en los campos de la aplicación; en el mejor de los casos, la persona que digitaba los datos, entendía e interpretaba adecuadamente lo escrito en el formato o formulario. [6]

En ocasiones la misma persona que realizaba el diagnóstico al vehículo, se encargaba de llenar el formato e interpretar y digitar los datos en la aplicación; aun así, esto podía ocasionar errores y tardanzas. Finalmente, se entrega la cotización al cliente, bien presentada, con los costos de la reparación. En la Figura 2.2 se muestra la cadena del proceso.

Figura 2.2 Procesos no soportados por la computación móvil



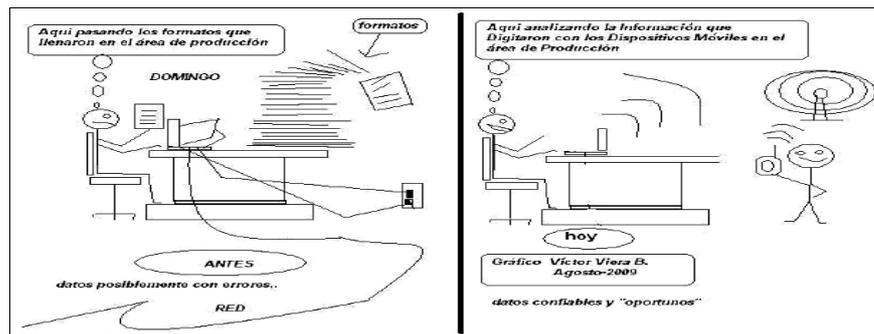
Fuente: [6]

Se llena un formulario; este se interpreta por otro usuario, y este usuario digita en la aplicación lo interpretado.

En la Figura 2.3 lado izquierdo, se observa la cantidad de formatos atrasados y pendientes por interpretar y digitar en la aplicación; el usuario debe sacrificar gran parte de tiempo para cumplir con las tareas no terminadas. En el lado derecho se observa cómo se evita los procesos o

reproceso de interpretación y digitación de los datos escritos en los formatos o formularios; aquí los datos ya han sido introducidos en la aplicación en el lugar donde ocurre el proceso; y mejor aún, los datos fueron digitados y enviados por medio de un dispositivo móvil.

Figura 2.3 Paradigma de computación móvil



Fuente: [6]

El lado izquierdo presenta cables, formatos y muchos errores de interpretación al centralizar el ingreso de datos mientras que el lado derecho representa agilidad en procesos y ahorro de recursos.

2.4.2 Dispositivo móvil

El dispositivo móvil realiza el proceso básico de recolección y envío de datos hacia la aplicación central. El dispositivo móvil contiene algunos procesos importantes; pero los procesos de mayor complejidad, en lo posible, se deben realizar en la aplicación central.

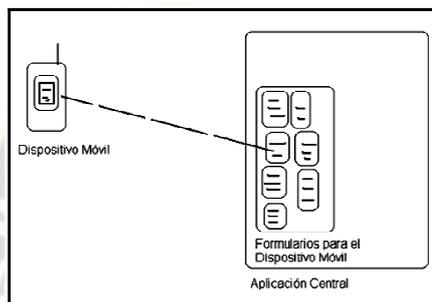
En el segundo escenario participan dos elementos:

- El dispositivo móvil.
- La aplicación central.

También se pueden encontrar sistemas computacionales móviles donde los datos no pasan por un proceso de sincronización propiamente dicho; estas aplicaciones poseen las validaciones necesarias en formularios o ventanas que se encuentran en el dispositivo; estas aplicaciones tienen la característica que están conectadas permanentemente a la aplicación central; se puede afirmar que la aplicación en el dispositivo móvil no funciona sin estar conectado o en línea con la aplicación central.

Otra particularidad de estos sistemas hace referencia a los formularios que se visualizarán en el dispositivo móvil, estarán ubicados en la aplicación central, este caso se presenta cuando el dispositivo móvil posee o tiene la capacidad de interpretar un formulario Web por medio de un navegador. Ver Figura 2.4.

Figura 2.4 Formularios ubicados en la aplicación central



Fuente: [6]

Adicionalmente, en este tipo de escenario, conectado, puede darse el caso de que se cree una aplicación para el dispositivo móvil, y esta aplicación contenga las ventanas que hacen posible la captura y posterior envío de datos a la aplicación central. En este escenario, las ventanas están almacenadas en el dispositivo, y estas se comunican con procesos de la aplicación central.

2.4.3 Sistema operativo móvil

Los Sistemas Operativos para teléfonos móviles se vuelven cada día más importantes debido a que la tecnología avanza y en materia de comunicaciones aún más, la telefonía celular cada vez se convierte más en una parte importante de nuestra sociedad, y en una sociedad que exige más y más, es importante diseñar sistemas que soporten las aplicaciones que se demandan, que sean fluidos, fáciles, accesibles y hasta divertidos. [6]

Es por eso que las compañías móviles han desarrollado una competencia bastante reñida en cuanto al desarrollo de sistema operativo se refiere, desde los inicios en los años 90 con las versiones de EPOC32 para PDA's hasta los más avanzados y sofisticados como Android, que además de ser eficientes y estables son multiplataforma, lo que hace que cualquier persona tenga acceso a ellos desde un celular básico hasta un Smartphone. [6]

Además cada vez más usuarios les agradan la idea de manipular y estilizar sus equipos y es lo que los nuevos sistemas operativos están ofreciendo y esto implica más retos de programación e incluso en el hardware.

2.5 Metodología de desarrollo Mobile D

Fue realizado, principalmente, por investigadores de la VTT (Instituto de Investigación Finlandés) y, a pesar de que es un método antiguo, sigue en vigor.[7]

El objetivo es conseguir ciclos de desarrollos muy rápidos en equipos muy pequeños (de no más de diez desarrolladores) trabajando en un mismo espacio físico. Según este método, trabajando de esa manera se deben conseguir productos totalmente funcionales en menos de diez semanas.

Se trata de un método basado en soluciones conocidas y consolidadas: Extreme Programming (XP), Crystal Methodologies y Rational Unified Process (RUP), XP para las prácticas de desarrollo, Crystal para escalar los métodos y RUP como base en el diseño de ciclo de vida.[7]

Figura 2.5 Ciclo de desarrollo Mobile-D



Fuente: [7]

Cada fase (excepto la inicial) tiene siempre un día de planificación y otro de entrega.

2.5.1 Elementos

Los elementos de la metodología Mobile-D se refieren a:[7]

Exploración. La fase de exploración, siendo ligeramente diferente del proceso de producción, se dedica al establecimiento de un plan de proyecto y los conceptos básicos, por lo tanto se puede

separar del ciclo principal de desarrollo. Los autores de la metodología ponen además especial atención a la participación de los clientes en esta fase.[7]

Inicialización. Durante esta fase, se preparan e identifican todos los recursos necesarios. Se preparan los planes para las siguientes fases y se establece el entorno técnico. Los autores de Mobile-D afirman que su contribución al desarrollo ágil se centra fundamentalmente en esta fase, en la investigación de la línea arquitectónica. Esta acción se lleva a cabo durante el día de planificación. Los desarrolladores analizan el conocimiento y los patrones arquitectónicos utilizados en la empresa y los relacionan con el proyecto actual. Se agregan las observaciones, se identifican similitudes y se extraen soluciones viables para su aplicación en el proyecto. Finalmente la metodología también contempla algunas funcionalidades nucleares que se desarrollan en esta fase, durante el día de trabajo.[7]

Productización. Llamada también fase de producto, se repite la programación de tres días (planificación-trabajo-liberación) se repite iterativamente hasta implementar todas las funcionalidades. Primero se planifica la iteración de trabajo en términos de requisitos y tareas a realizar. Se preparan las pruebas de la iteración de antemano (es por ello el nombre de esta técnica de Test Driven Development, TDD). Las tareas se llevaran a cabo durante el día de trabajo. Durante el último día se lleva a cabo la integración del sistema seguida de las pruebas de aceptación.[7]

Estabilización. Se llevan a cabo las últimas acciones de integración para asegurar que el sistema completo funciona correctamente. Esta será la fase más importante en los proyectos multi-equipo con diferentes subsistemas desarrollados por equipos distintos. En esta fase, los desarrolladores realizaran tareas similares en las que debían desarrollar en la fase de “productización”, aunque en este caso todo el esfuerzo se dirige a la integración del sistema. Adicionalmente se puede considerar en esta fase la producción de la documentación.[7]

Pruebas y Reparación. La última fase de prueba y reparación del sistema tiene como objetivo la disponibilidad de una versión estable y plenamente funcional del sistema. El producto terminado e integrado se prueba con los requisitos de cliente y se eliminan todos los defectos encontrados.[7]

2.6 Geolocalización

La geolocalización es la capacidad de asignar coordenadas geográficas a la información por medio de herramientas informáticas. La generalización de la tecnología GPS en dispositivos de uso personal como los teléfonos móviles y ordenadores personales ha permitido que esta capacidad esté al alcance de cualquier ciudadano, y como consecuencia, el desarrollo de aplicaciones a distintos campos. [8]

La geolocalización tiene múltiples aplicaciones, pero las que aquí nos ocupan son las relacionadas con las redes sociales y la tendencia actual de compartir distintos aspectos de cada vida a través de las nuevas tecnologías. [8]

En los últimos años, el “hecho social” se ha convertido en una tendencia y también en el santo y seña de una generación (principalmente los nacidos después de 1990) que ha redefinido conceptos como intimidad, privacidad, relación y red social.

La geolocalización es un desarrollo lógico para la interacción social en Internet y las redes sociales, ya que contrariamente a lo que sectores de la población menos familiarizados con este entorno creen, las redes sociales no son mundos virtuales ajenos al mundo “real”, sino que por el contrario una de sus características es la integración, la involución: no se puede hablar de éxito de una red social hasta que ésta consiga trascender la propia red y convertirse en parte de la vida diaria “real”. [8]

Si se unen los sistemas de GPS con los móviles y los *Social Media*, generando la posibilidad de comunicar y compartir el lugar concreto en el que se ubica una persona en cada momento, nace un concepto nuevo llamado “Geolocalización social”. [8]

Por tanto, la geolocalización social hace referencia a las nuevas formas de relación social que surgen gracias a la geolocalización de los individuos con sus móviles y que pueden desarrollarse mediante diversas herramientas, que son básicamente cuatro: Facebook Places, Google Local, Foursquare y Yelp. [8]

2.7 API Google Maps

Google Maps es un servidor de aplicaciones de mapas en la web que pertenece a Alphabet Inc. Ofrece imágenes de mapas desplazables, así como fotografías por satélite del mundo e incluso la ruta entre diferentes ubicaciones o imágenes a pie de calle con Google Street View. [9]

Existe una variante a nivel entorno de escritorio llamada Google Earth que ofrece Alphabet Inc. también de forma gratuita. En 2014, los documentos filtrados por Edward Snowden revelaron que Google Maps es parte y víctima del entramado de vigilancia mundial operado por varias agencias de inteligencia occidentales y empresas tecnológicas.[9]

Como en las aplicaciones web de Google, se usan un gran número de archivos Javascript para crear Google Maps permitiendo que el usuario puede mover el mapa, la visualización del mismo se baja desde el servidor. Cuando un usuario busca un negocio, la ubicación es marcada por un indicador en forma de pin, el cual es una imagen PNG transparente sobre el mapa. Para lograr la conectividad sin sincronía con el servidor, Google aplicó el uso de AJAX dentro de esta aplicación. Es una aplicación para el desarrollo de mapas.[9]

Todas las aplicaciones de Google Maps JavaScript API requieren autenticación.[10]

- Usuarios de la API estándar: si usa la API con el plan estándar, debe emplear una clave de API configurada en el proyecto que determine.
- Usuarios de la Premium Plan: si usa la API con el Google Maps APIs Premium Plan, existen dos métodos de autenticación.
 - Usa una clave de API configurada en el proyecto de Google Maps APIs Premium Plan.
 - Usa tu ID de cliente en lugar de la clave de API.

Para la autenticación para la API estándar se debe obtener una clave de API.

La clave de API permite controlar el uso de la API por parte de la aplicación en Google API Console y para comenzar a usar Google Maps JavaScript API, según [10], se debe realizar el proceso siguiente:

- a) Ingresar a Google API Console.
- b) Crear un proyecto.

- c) Seleccionar opción Continue para habilitar la API y cualquier servicio relacionado.
- d) De la página Credentials, obtener una clave de API (y configurar las restricciones para esta).

Para especificar la clave, incluir como valor del parámetro key cuando se cargue la API.[10]

Las instrucciones HTML para cargar el mapa se presenta a continuación:

```
<script async defer  
src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap"  
type="text/javascript">  
</script>
```

2.8 Modelo Vista Controlador

Las aplicaciones PHP bien escritas siguen el patrón de diseño de software MVC (Modelo-Vista-Controlador). Programar utilizando MVC consiste en separar la aplicación en tres partes principales. El modelo representa los datos de la aplicación, la vista hace una presentación del modelo de datos, y el controlador maneja y enruta las peticiones [requests] hechas por los usuarios.[11]

Por tanto:

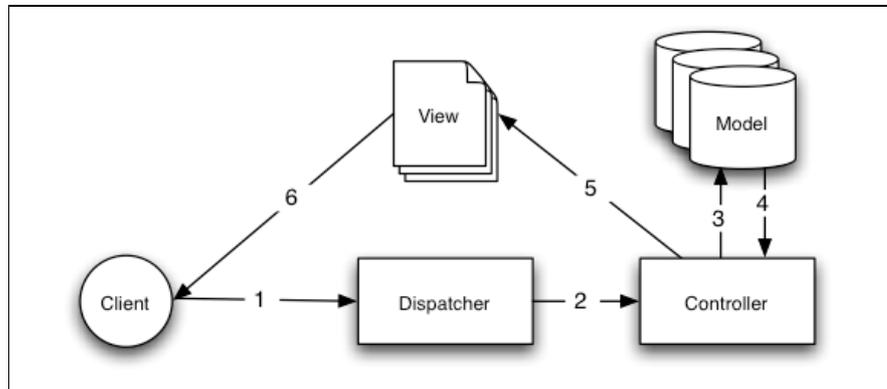
El **Modelo** encargado de la lógica de la aplicación y la persistencia de los datos.

Las **Vistas** son las responsables de mostrar al usuario el resultado que obtienen del modelo a través del controlador.

El **Controlador** encargado de gestionar las peticiones del usuario, procesarlas invocando al modelo y mostrarlas al usuario a través de las vistas.

Como ejemplo del funcionamiento de este patrón, la figura siguiente muestra un diseño sencillo de una petición [request] MVC en PHP, a efectos ilustrativos, se asume que un usuario llamado Ricardo acaba de hacer clic en el enlace “¡Comprar un pastel personalizado ahora!” de la página inicial de la aplicación. (Ver figura 2.6)

Figura 2.6 Modelo vista controlador



Fuente: [11]

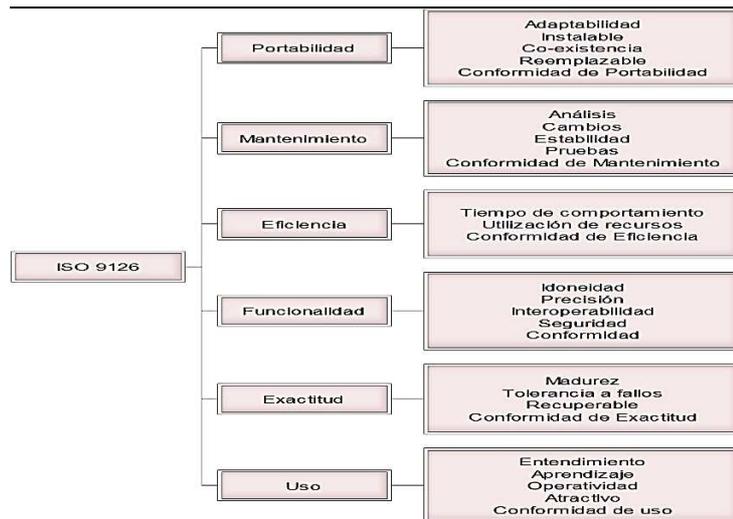
- Ricardo hace clic en el enlace apuntando a <http://www.ejemplo.com/pasteles/comprar>, y su navegador hace una petición al servidor web.
- El despachador comprueba la URL de la petición (/pasteles/comprar), y le pasa la petición al controlador adecuado.
- El controlador realiza lógica de aplicación específica. Por ejemplo, puede comprobar si Ricardo ha iniciado sesión.
- El controlador también utiliza modelos para acceder a los datos de la aplicación. La mayoría de las veces los modelos representan tablas de una base de datos, aunque también pueden representar entradas LDAP, canales RSS, o ficheros en el sistema.
- Una vez que el controlador ha hecho su trabajo en los datos, se pasa a la vista. La vista toma los datos y los deja listos para su presentación al usuario en formato HTML, PDF, XML o un objeto JSON.
- Una vez que el objeto encargado de procesar vistas en PHP ha utilizado los datos del controlador para construir una vista completa, el contenido se devuelve al navegador de Ricardo.

2.9 Calidad de software

Se detallarán los criterios de calidad de ISO 9126, respecto a las Métricas de Calidad del Producto de Software. ISO 9126 presenta seis métricas de evaluación de software:

Funcionalidad, Fiabilidad, Usabilidad, Eficiencia, Mantenibilidad y Portabilidad; las que a su vez se dividen en otros atributos de las cuales solamente serán evaluadas una por cada grupo.

Figura 2.7 Factores de evaluación de calidad



Fuente:[12]

Este proyecto abarca el estudio de cuatro aspectos de la calidad establecidas por la ISO 9126.

- ❖ Métricas de Funcionalidad
- ❖ Métricas de Usabilidad
- ❖ Métricas de Portabilidad
- ❖ Métricas de Mantenibilidad

➤ **Métricas de Funcionalidad**

Métrica para medir que si las funciones satisfacen necesidades declaradas o implícitas [ISO 9126, 1991], la que a su vez indica que se deben considerar los siguientes atributos:

Adecuidad

Como se muestra en la 2.2, se consideran los factores para medir la Adecuidad de un sistema.

Nombre: Adecuidad	
Propósito:	Qué tan completa está la implementación funcional.
Método de aplicación:	Contar las funciones faltantes detectadas en la evaluación y comparar con el número de funciones descritas en la especificación de requisitos.
Medición, fórmula:	$X = 1 - A/B$ A = número de funciones faltantes B = número de funciones descritas en la especificación de requisitos
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, más completa.
Tipo de escala:	Absoluta
Tipo de medida:	X = count/count A = count B = count
Fuente de medición:	Especificación de requisitos Diseño Código fuente Informe de revisión
ISO/IEC 12207 SLCP:	Validación Revisión conjunta
Audiencia:	Requeridores Desarrolladores

Tabla 2.2 Métricas de adecuadad

Fuente: [12]

➤ Métricas de Usabilidad

Se define la usabilidad como la capacidad de un producto software de ser comprendido, aprendido, usado y de ser atractivo para el usuario, en condiciones específicas de uso [ISO 9126]. Se debe tener en cuenta que la usabilidad no depende sólo del producto, sino también del usuario, para ello se define las siguientes métricas:

Entendibilidad

La tabla siguiente refleja que la entendibilidad está en función de número de funciones como proporción sobre el número total de funciones.

Nombre:	Entendibilidad
Propósito:	Qué proporción de las funciones del sistema son evidentes al usuario.
Método de aplicación:	Contar las funciones evidentes al usuario y comparar con el número total de funciones.
Medición, fórmula:	$X = A/B$ A = número de funciones (o tipos de funciones) evidentes al usuario B = total de funciones (o tipos de funciones)
Interpretación:	$0 \leq X \leq 1$ Entre más cercano a 1, mejor.
Tipo de escala:	Absoluta
Tipo de medida:	$X = \text{count}/\text{count}$ A = count B = count
Fuente de medición:	Especificación de requisitos Diseño Informe de revisión
ISO/IEC 12207 SLCP:	Verificación Revisión conjunta
Audiencia:	Requeridores Desarrolladores

Tabla 2.3 Métrica de entendibilidad

Fuente: [12]

➤ Métricas de Portabilidad

Es la capacidad del conjunto de atributos relacionados con la capacidad de un sistema de software para ser transferido y adaptado desde una plataforma a otra, por ejemplo de ser ejecutado en Windows o en otra plataforma, por tanto deben ser ejecutadas en diversas plataformas y si pueden ejecutarse sin problemas o alteraciones de código.

Conformidad de la portabilidad

La tabla 2.4, establece la portabilidad del sistema y determina en que plataformas trabaja el software ya sea con mínimos cambios o ninguna alteración.

Nombre:	Conformidad de Portabilidad
Propósito:	Qué tan conforme es la portabilidad del producto con regulaciones, estándares y convenciones aplicables.
Método de aplicación:	Contar los artículos encontrados que requieren conformidad y comparar con el número de artículos en la especificación que requieren conformidad.
Medición, fórmula:	$X = A/B$ A = número de artículos implementados de conformidad B = total de artículos que requieren conformidad
Interpretación:	$0 \leq X \leq 1$. Entre más cercano a 1, más completa.
Tipo de escala:	Absoluta
Tipo de medida:	$X = \text{count}/\text{count}$. Dónde: A = count; B = count
Fuente de medición:	Especificación de conformidad y estándares, convenciones y regulaciones relacionados. Diseño , Código fuente, Informe de revisión
ISO/IEC 12207 SLCP:	Verificación Revisión conjunta
Audiencia:	Requeridores, desarrolladores

Tabla 2.4 Métrica de portabilidad

Fuente: [12]

➤ Métricas de Mantenibilidad

Este atributo mide la capacidad de un sistema para resistir ataques (tanto accidentales como intencionados) contra su seguridad. En este grupo, se hace referencia en particular a la integridad del software, considerando dos atributos: amenaza y seguridad.

Amenaza es la probabilidad de que un ataque de un tipo determinado ocurra en un tiempo determinado. La *seguridad* es la probabilidad de que se pueda repeler el ataque de un tipo determinado. [5]

La integridad del sistema se define como:

$$\text{Integridad} = \Sigma [(1 - \text{amenaza}) \times (1 - \text{seguridad})]$$

Donde se suman la cantidad de amenazas y los medios de seguridad para cada tipo de ataque.

2.10 Estimación de Costos COCOMO II

El modelo constructivo de Costo (o COCOMO, del inglés COConstructive COst MOdel) es un modelo matemático de base empírica utilizado para estimación de costos para el desarrollo de un software.

COCOMO II está compuesto por tres modelos denominados: Composición de Aplicación, Diseño Temprano y Post-Arquitectura.

Éstos surgen en respuesta a la diversidad del mercado actual y futuro de desarrollo de software. Esta diversidad se presenta en la tabla siguiente.

Aplicaciones desarrolladas por usuarios finales			
Generadores de aplicaciones	de	Aplicaciones con componentes	Sistemas integrados
Infraestructura			

Tabla 2.5 Distribución del mercado de software

Fuente: [13]

- **Aplicaciones desarrolladas por usuarios finales:** En este sector se encuentran las aplicaciones de procesamiento de información generadas directamente por usuarios finales, mediante la utilización de generadores de aplicaciones tales como planillas de cálculo, sistemas de consultas, etc. Estas aplicaciones surgen debido al uso masivo de estas herramientas, conjuntamente con la presión actual para obtener soluciones rápidas y flexibles.
- **Generadores de Aplicaciones:** En este sector operan firmas como Lotus, Microsoft, Novell, Borland con el objetivo de crear módulos pre-empaquetados que serán usados por usuarios finales y programadores.
- **Aplicaciones con Componentes:** Sector en el que se encuentran aquellas aplicaciones que son específicas para ser resueltas por soluciones pre-empaquetadas, pero son lo suficientemente simples para ser construidas a partir de componentes interoperables. Componentes típicas son constructores de interfaces gráficas, administradores de bases de datos, buscadores inteligentes de datos, componentes de dominio-específico (medicina,

finanzas, procesos industriales, etc.). Estas aplicaciones son generadas por un equipo reducido de personas, en pocas semanas o meses.

- **Sistemas Integrados:** Sistemas de gran escala, con un alto grado de integración entre sus componentes, sin antecedentes en el mercado que se puedan tomar como base. Porciones de estos sistemas pueden ser desarrolladas a través de la composición de aplicaciones. Entre las empresas que desarrollan software representativo de este sector, se encuentran grandes firmas que desarrollan software de telecomunicaciones, sistemas de información corporativos, sistemas de control de fabricación, etc.
- **Infraestructura:** Área que comprende el desarrollo de sistemas operativos, protocolos de redes, sistemas administradores de bases de datos, etc. Incrementalmente este sector direccionará sus soluciones, hacia problemas genéricos de procesamiento distribuido y procesamiento de transacciones, a soluciones middleware. Firmas representativas son Microsoft, Oracle, SyBase, Novell y NeXT.

Los tres modelos de COCOMO II se adaptan tanto a las necesidades de los diferentes sectores descritos, como al tipo y cantidad de información disponible en cada etapa del ciclo de vida de desarrollo, lo que se conoce por granularidad de la información.

Se puede afirmar que para las aplicaciones desarrolladas por usuarios finales no se justifica la utilización de un modelo de estimación de costos. Estas aplicaciones normalmente se construyen en poco tiempo, por lo tanto requieren solamente una estimación basada en actividades.

El modelo Composición de Aplicación, es el modelo de estimación utilizado en los proyectos de software que se construyen a partir de componentes pre-empaquetadas. En este caso, se emplean Puntos Objeto para estimar el tamaño del software, lo cual está acorde al nivel de información que generalmente se tiene en la etapa de planificación, y el nivel de precisión requerido en la estimación de proyectos de esta naturaleza.

Para los demás sectores del mercado se aplica un modelo mixto, combinación de los tres modelos.

El modelo Composición de Aplicación se emplea en desarrollos de software durante la etapa de prototipación.

El modelo Diseño Temprano se utiliza en las primeras etapas del desarrollo en las cuales se evalúan las alternativas de hardware y software de un proyecto. En estas etapas se tiene poca información, lo que concuerda con el uso de Puntos Función, para estimar tamaño y el uso de un número reducido de factores de costo.

El modelo Post-Arquitectura se aplica en la etapa de desarrollo propiamente dicho, después que se define la arquitectura del sistema, y en la etapa de mantenimiento. Este modelo utiliza:

- Puntos Función y/o Líneas de Código Fuente para estimar tamaño, con modificadores que contemplan el reuso, con y sin traducción automática, y el "desperdicio"
- Un conjunto de 17 atributos, denominados factores de costo, que permiten considerar características del proyecto referentes al personal, plataforma de desarrollo, etc., que tienen injerencia en los costos.
- Cinco factores que determinan un exponente, que incorpora al modelo el concepto de deseconomía y economía de escala. Estos factores reemplazan los modos Orgánico, Semiacoplado y Empotrado del modelo COCOMO '81.

2.10.1 Estimación de esfuerzo

El esfuerzo necesario para concretar un proyecto de desarrollo de software, cualquiera sea el modelo empleado, se expresa en meses/persona (PM) y representa los meses de trabajo de una persona a tiempo completo, requeridos para desarrollar el proyecto.

2.10.2 Modelo de composición de aplicación

La fórmula propuesta en este modelo es la siguiente:

$$PM = NOP / PROD$$

Dónde:

NOP (Nuevos Puntos Objeto): Tamaño del nuevo software a desarrollar expresado en

Puntos Objeto y se calcula de la siguiente manera:

$$NOP = OP \times (100 - \%reuso)/100$$

OP (Puntos Objeto): Tamaño del software a desarrollar expresado en Puntos Objeto

%reuso: Porcentaje de reuso que se espera lograr en el proyecto

PROD: Es la productividad promedio determinada a partir del análisis de datos de proyectos.





CAPITULO III MARCO PRACTICO

3 MARCO PRÁCTICO

3.1 Introducción

Este capítulo trata sobre el desarrollo de la aplicación móvil siguiendo para ello los lineamientos de la metodología Mobile-D, ajustada a las necesidades y requerimientos de los usuarios identificados.

El ciclo del proyecto se divide en cinco fases: exploración, inicialización, productización, estabilización y prueba del sistema como se muestra en la siguiente figura.

Figura 3.1 Fases de Mobile-D



Fuente: [7]

3.2 Exploración

La fase de exploración trata principalmente sobre un análisis previo de la situación e identificación de las necesidades que se deben resolver y para ello se realizan principalmente entrevistas y encuestas.

3.2.1 Recopilación de información

La recopilación de datos de acuerdo a la metodología requiere en principio establecer reuniones con las personas que tienen el interés. Se realizó principalmente entrevistas a trece personas que padecen de convulsiones de epilepsia, quienes formularon las necesidades que se requiere resolver.

Los problemas detectados fueron analizados a partir de un cuestionario respondido por personas que sufren de estos ataques de epilepsia.

Cuestionario:

- a) Usted cuando sufre de convulsiones, tiene alguna persona que le preste ayuda?

- b) Sufre de convulsiones de manera frecuente?
- c) La recuperación de su salud es inmediata?
- d) El historial clínico está a disposición de usted y del especialista médico?
- e) Dispone de medios de ayuda inmediata?
- f) Acude al médico en busca de apoyo?
- g) Usted sabe cuándo esta por sufrir un ataque compulsivo?
- h) Los médicos conocen su situación médica de manera confiable?

Los resultados tabulados se presentan a continuación:

- a) Usted cuando sufre de convulsiones, tiene alguna persona que le preste ayuda?

Si	No
2	11

- b) Sufre de convulsiones de manera frecuente?

Si	No
9	4

- c) La recuperación de su salud es inmediata?

Si	No
1	12

- d) El historial clínico está a disposición de usted y del especialista médico?

Si	No
1	12

- e) Dispone de medios de ayuda inmediata?

Si	No
1	12

- f) Acude al médico en busca de apoyo?

Si	No

g) Usted sabe cuándo esta por sufrir un ataque compulsivo?

5	8
---	---

h) Los médicos conocen su situación médica de manera confiable?

Si	No
2	11

Si	No
3	10

Entonces, a partir de los datos recopilados, se procede a la elaboración de un resumen de los resultados y generando un cuadro relativo porcentual como se muestra en la tabla 3.1

Tabla 3.1 Resultados de encuesta

Pregunta	SI (%)	NO (%)
1	15.4	84.6
2	30.8	69.2
3	7.7	92.3
4	7.7	92.3
5	7.7	92.3
6	38.5	61.5
7	15.4	84.6
8	23.1	76.9
TOTALES	18.3	81.7

Fuente: Elaboración propia

Según los resultados obtenidos, se observa claramente que la necesidad de desarrollo de una aplicación móvil para apoyar y solicitar ayuda médica es realmente necesaria y se requiere con 81% de necesidad.

3.2.2 Identificación de actores

Según las investigaciones realizadas, se tienen los actores: administrador y usuario.

Administrador

Es responsable de monitorear el sistema y que este no presente fallas.

Está a cargo de la aplicación web donde mediante un mapa del Google Maps, la ubicación por donde la persona presenta problemas de salud, siempre y cuando el usuario haya mandado la solicitud de ayuda inmediata respectiva desde la aplicación móvil.

El administrador dispone de una interface web que presenta información de los pacientes que hacen uso de la aplicación móvil además de la generación de estadísticas individuales y grupales, presentando una además los puntos geográficos donde se presentan los problemas que las personas con convulsiones presentan.

En el futuro, el administrador será personal de instituciones encargadas de la salud y familiares principalmente.

Usuario

Es la persona que se registra e inicia la aplicación de alerta a los familiares y centro de salud.

El usuario debe en principio descargar la aplicación móvil desde la tienda del Google Play, donde estará disponible la app para que pueda ser instalado en los dispositivos móviles.

Luego de haber instalado la aplicación móvil, debe registrarse introduciendo los datos personales del usuario además de introducir los números de teléfono celular a los cuales se enviarán alertas y mensajes de ayuda inmediata ya sea por llamadas telefónicas o utilizando aplicaciones de comunicación gratuitas.

3.2.3 Requerimientos

Mediante el análisis de requerimientos se tiene un mecanismo para identificar lo que requiere el usuario (en este caso la persona que sufre las convulsiones), especificando necesidades y de esa manera establecer una estrategia de solución, además que sirven de apoyo para el diseño y desarrollo del sistema.

Esta información será plasmada en principio como requerimientos iniciales y luego tras una reunión, se deben identificar aquellos requerimientos que realmente serán implementadas en el desarrollo de la aplicación. Los requerimientos se convierten en historias de usuario, de desarrolla tal requerimiento y posteriormente tras la prueba de aceptación se debe rechazar o aceptar el trabajo desarrollado.

ADMINISTRADOR

Esta historia está referida a la necesidad de reflejar de manera gráfica los puntos geográficos donde se generan las llamadas de solicitud de ayuda de parte de los usuarios que sufren ataques de epilepsia.

Historia de usuario		23/Abril/2016
Historia de usuario N° 001		
Usuario: Administrador.		
Nombre: Información		Prioridad: Alta
Descripción	Se debe mostrar de manera gráfica utilizando un mapa el recorrido en tiempo real de la persona a la que se quiere monitorear en el instante en que se presente las convulsiones. Este monitoreo estará en base a las coordenadas geográficas que envía el dispositivo Smartphone cada cinco minutos. Los datos serán almacenados en el mismo dispositivo o bien en un servidor de base de datos remota.	
Estimación	Se estima desarrollar la aplicación en un periodo no mayor a 5 días.	

Tabla 3.2 Historia de usuario 1
Fuente: Elaboración Propia

Prueba de aceptación	
	Mostrar de forma gráfica en un mapa georreferenciado el lugar donde se presente el problema de salud.

Tabla 3.3 Prueba de aceptación: Información
Fuente: Elaboración Propia

USUARIO

El usuario para ser considerado en el sistema web, debe registrar los datos personales tanto personales como de los dos contactos a los que se deben hacer las llamadas de solicitud de ayuda cuando se presenten las circunstancias.

Historia de usuario		23/Abril/2016
Historia de usuario N° 002: Registro de usuarios		
Nombre: Información		Prioridad: Alta
Descripción	Se debe registrar a la persona que sufre de ataques de epilepsia, este registro debe ser personal mediante la aplicación móvil. Adicionalmente se deben registrar una serie de número telefónico a los que se debe comunicar de inmediato el problema de salud que tenga el propietario del dispositivo inteligente.	
Estimación	Diez días de trabajo	

Tabla 3.4 Historia de usuario 2
Fuente: Elaboración Propia

Prueba de aceptación	
Descripción	El sistema tiene registrado los datos del usuario. El sistema tiene registrado los números de teléfono a los que enviara el mensaje de alerta.

Tabla 3.5 Prueba de aceptación Registro de usuarios
Fuente: Elaboración Propia

USUARIO

La ayuda debe ser inmediata tan solo presionando un botón en el dispositivo móvil

Historia de usuario		23/Abril/2016
Historia de usuario N° 003		
Nombre: Alerta		Prioridad: Media
Descripción	Quiero que la aplicación refleje de manera inmediata apretando un botón en el dispositivo inteligente y comunique la ubicación geográfica donde me encuentro. Adicionalmente quiero enviar alertas a los centros de salud o bien a una empresa de radiotaxi para recibir auxilio médico de manera inmediata.	
Estimación	Diez días para el desarrollo	

Tabla 3.6 Historia de usuario 3
Fuente: Elaboración Propia

Prueba de aceptación	
Descripción	El sistema debe tener un módulo de alerta inmediata y comunique la situación de emergencia tanto a la administración del sistema como a celulares de los familiares.

Tabla 3.7 Prueba de aceptación: Alerta
Fuente: Elaboración propia

USUARIO

La solicitud de utilizar la cámara del dispositivo web para guardar un video sobre el entorno que se desarrolla cerca de la persona que sufre el ataque de epilepsia.

Historia de usuario		23/Abril/2016
Historia de usuario N° 004		
Nombre: Video	Prioridad: Media	
Descripción	La aplicación debe guardar mediante la cámara un video sobre la situación donde se encuentra el paciente y de esta manera registrar de manera más adecuada cómo se va desarrollando la situación problemática por la cual atraviesa el paciente.	
Estimación	Diez días para el desarrollo	

Tabla 3.8 Historia de usuario 4
Fuente: Elaboración Propia

Prueba de aceptación	
Descripción	El sistema registra sonido e imagen en la base de datos.

Tabla 3.9 Prueba de aceptación: Video
Fuente: Elaboración propia

USUARIO

La aplicación debe ser capaz de generar estadísticas sobre los casos registrados de epilepsia tanto por usuario así como por periodos de tiempo y ser mostrados en la pantalla de la aplicación web.

Historia de usuario		23/Abril/2016
Historia de usuario N° 005		
Nombre: Generación de estadísticas.		Prioridad: Media
Descripción	<p>La aplicación debe tener disponible módulos para la generación de estadísticas sobre las convulsiones que sufren los pacientes desde la interface web.</p> <p>Podrá generar estadísticas por un paciente o de manera global, incluso por periodos de tiempo.</p>	
Estimación	Diez días para el desarrollo	

Tabla 3.10 Historia de usuario 5
Fuente: Elaboración Propia

Prueba de aceptación	
Descripción	<p>Estadísticas generadas por:</p> <p>Cada paciente.</p> <p>Listado de pacientes.</p> <p>Ataques sufridos por cada paciente.</p> <p>Considerar periodos de tiempo.</p>

Tabla 3.11 Prueba de aceptación: Estadísticas
Fuente: Elaboración propia

USUARIO

Historia de usuario		23/Abril/2016
Historia de usuario N° 006		
Nombre: Consultas.		Prioridad: Media
Descripción	<p>Los usuarios deben estar en condiciones de realizar las consultas desde los dispositivos móviles así como desde la aplicación web.</p> <p>Esa consulta debe ser respondida por el servidor tanto en forma gráfica usando el Google Maps o bien mostrando los resultados en forma de tabla.</p>	
Estimación	Diez días para el desarrollo	

Tabla 3.12 Historia de usuario 6
Fuente: Elaboración Propia

Prueba de aceptación	
Descripción	Generación de consultas desde la aplicación móvil o bien desde la aplicación web.

Tabla 3.13 Consultas
Fuente: Elaboración propia

3.3 Inicialización

Esta fase a partir de las necesidades identificadas como requerimientos del sistema, se elabora un resumen de los requerimientos formales que serán implementados tanto en la aplicación móvil así como la aplicación web.

3.3.1 Requerimientos del sistema

Referencia	Descripción del requerimiento	Tipo
R1W	La aplicación debe ser capaz de monitorear cada periodo de tiempo la posición geográfica del usuario.	Funcional
R2W	Los puntos geográficos donde se presentan los accidentes estarán almacenados en una base de datos. Se requiere de un dispositivo con conexión a internet y GPS incluido.	Funcional
R3W	El administrador tiene acceso a toda la información almacenada en la base de datos.	Funcional
R4W	La aplicación desarrollada debe ajustarse de manera automática a diferentes dispositivos móviles inteligentes.	Funcional
R5W	El usuario debe registrarse en la aplicación móvil incluyendo los números de teléfono de los familiares, centros de salud y empresas de radio taxis, o cualquier institución o persona que esté dispuesta a prestar ayuda inmediata.	Funcional
R6W	Se deben validar todos los campos antes de adicionar, eliminar o modificar información de la base de datos.	Funcional
R7W	Consulta de usuarios	Funcional
R8W	Generación de Historial médico	Funcional
R9W	Compatibilidad con los navegadores.	No funcional
R10W	Seguridad para cuentas de usuario protegidos por contraseña	No funcional
R11W	Facilidad de uso, interfaz gráfica y navegación amigable al usuario	No funcional

Tabla 3.14 Tabla de Requerimientos
Fuente: Elaboración propia

Luego de haber elaborado las historias de usuario, se procede a clarificar cuáles serán los requerimientos que se deben cumplir a fin de lograr el objetivo, estos requerimientos se ven reflejados en la tabla anterior.

De acuerdo con la tabla de requerimientos, se tienen que cumplir cinco requerimientos plenamente identificados.

3.3.2 Requerimientos de la aplicación móvil

Respecto a la aplicación móvil, los requerimientos definidos para su desarrollo, se establecen los siguientes requisitos.

Referencia	Descripción del requerimiento	Tipo
R1M	Identificación de usuarios.	Funcional
R2M	Realizar registro de usuarios	Funcional
R3M	Realizar registro de contactos a ser comunicados ante emergencias.	Funcional
R4M	Consulta de ubicaciones geográficas.	Funcional
R5M	Consulta de usuarios y sus contactos.	Funcional

Tabla 3.15 Tabla de requerimientos de la aplicación móvil

Fuente: Elaboración propia

3.3.3 Requisitos del sistema

De acuerdo a los requerimientos que se detallaron anteriormente se debe realizar los casos de uso en base a los mismos, para de esa manera ver cuál va ser la interacción del usuario con el sistema según los requisitos.

Para el establecimiento de los diagramas de casos de uso, inicialmente se deben definir el caso de uso de alto nivel y a partir del mismo los casos de uso expandido. Un aspecto muy importante es el nivel de acceso que se presenta en el sistema, mismo que se puede diferenciar dos niveles de usuarios:

- Usuario administrador
- Usuario operador

El usuario administrador es la persona que tiene acceso completo al sistema, incluyendo al portal web.

El usuario operador, serán las personas que tienen acceso a la aplicación móvil donde serán registrados y los que utilicen y envíen mensajes de ayuda desde el dispositivo móvil.

3.3.4 Sistema propuesto

El sistema desarrollado está conformado por una aplicación centralizada cuyos accesos son a dos niveles: uno web y el otro desde los dispositivos móviles.

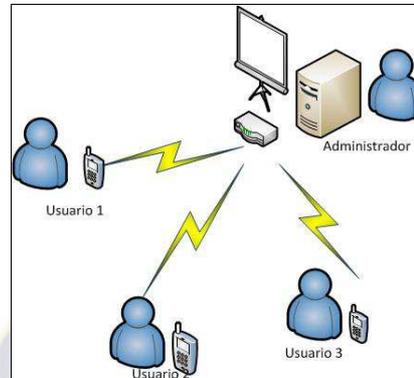
Figura 3.2 Arquitectura del sistema Epimovil



Fuente: Elaboración propia

Por otra parte, el subsistema móvil trabaja en base a un servidor web y servidor de base de datos, y mediante el canal inalámbrico, se establece la conexión del mismo con los dispositivos inteligentes como se muestra en la siguiente figura.

Figura 3.3 Arquitectura del sub-sistema móvil



Fuente: Elaboración propia

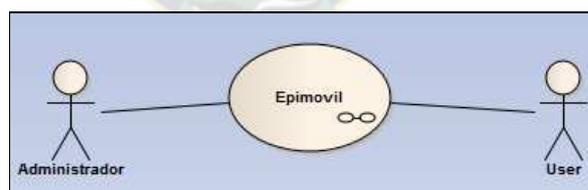
Como se observa, la aplicación central estará almacenada en un servidor web, mientras que las terminales serán los celulares inteligentes a cargo de las personas que hayan instalado la aplicación móvil.

3.3.5 Modelado de procesos

3.3.5.1 Diagramas de casos de uso

El modelo de contexto del sistema se presenta en la siguiente figura.

Figura 3.4 Diagrama de contexto del sistema



Fuente: Elaboración propia

Los actores interactúan con el sistema 'Epimovil' son el administrador y el usuario

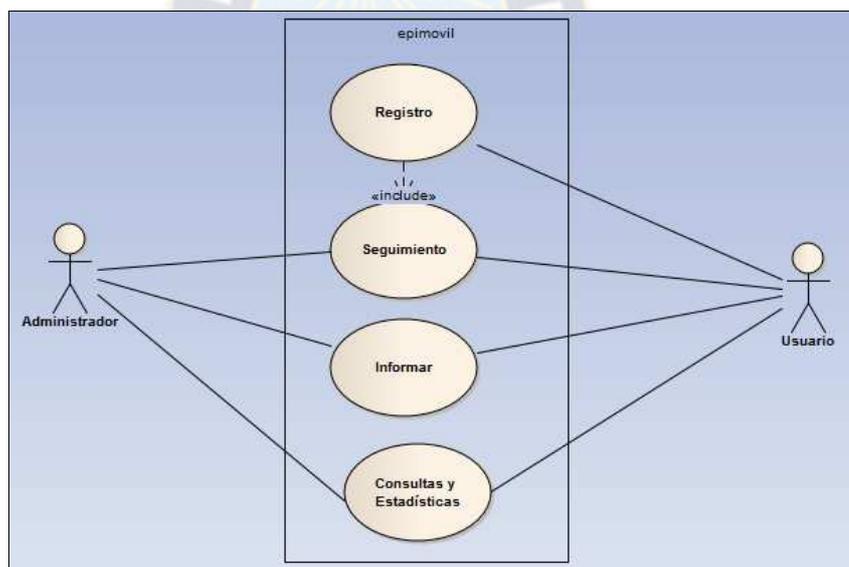
De forma interna, en Epimovil se tienen cuatro procesos identificados a ser desarrollados: Registro de usuario, en este caso es la persona que desde su celular debe registrarse incluyendo los números de celulares de sus familiares a los cuales se enviará el mensaje de solicitud de ayuda inmediata.

Seguimiento, una vez que el usuario haya registrado sus datos personales en el sistema Epimovil, la aplicación debe estar en ejecución permanente en el dispositivo, para que en el momento inoportuno pueda enviar el mensaje de ayuda.

Informar, proceso que envía un mensaje a los números de celulares registrados de los familiares. La aplicación también grabará el audio del entorno e imagen obtenida con la cámara del celular de forma transparente.

Consultas y estadísticas, procesos que generan información a partir de los datos almacenados en la base de datos.

Figura 3.5 Caso de uso general del sistema



Fuente: Elaboración propia

3.4 Productización

3.4.1 Planificación

Se plantea desarrollar tres iteraciones, cada una de ellas debe generar productos entregables y funcionales que conformarán una pila de productos que serán parte del producto final.

Para el desarrollo de la aplicación, el tiempo estimado será de sesenta días calendarios, es decir que cada iteración tiene un tiempo promedio de 20 días.

# H.U.	Propósito	Prioridad
1	Módulo de registro de usuario	Alta
2	Seguimiento continuo	Media
3	Alerta	Alta
4	Módulo de logueo de la persona.	Alta
5	Grabación de sonido ambiente	Alta
6	Fotografías	Alta
7	Envío de mensaje a celulares registrados.	Alta
8	Despliegue de datos	Alta

Tabla 3.16 Planificación de tareas

Fuente: Elaboración propia

Según las actividades planificadas para desarrollar el sistema, se tiene un conjunto de actividades detalladas que deben ser desarrolladas y que se presentan en el siguiente diagrama de Gantt.

3.4.2 Cronograma de trabajo

La siguiente figura establece el cronograma de trabajo de los tres sprints que serán realizados como parte del trabajo.

Sprint	Tarea	Duración
1	R1W	4
1	R2W	5
1	R3W	4
1	R4W	5
2	R5W	5
2	R6W	5
2	R7W	4
2	R8W	6
3	R1M	5
3	R2M	4
3	R3M	5
3	R4M	4
3	R5M	4

Tabla 3.17 Tareas y su duración

Fuente: Elaboración propia

Nota:

R1W representa el requerimiento 1 del desarrollo web.

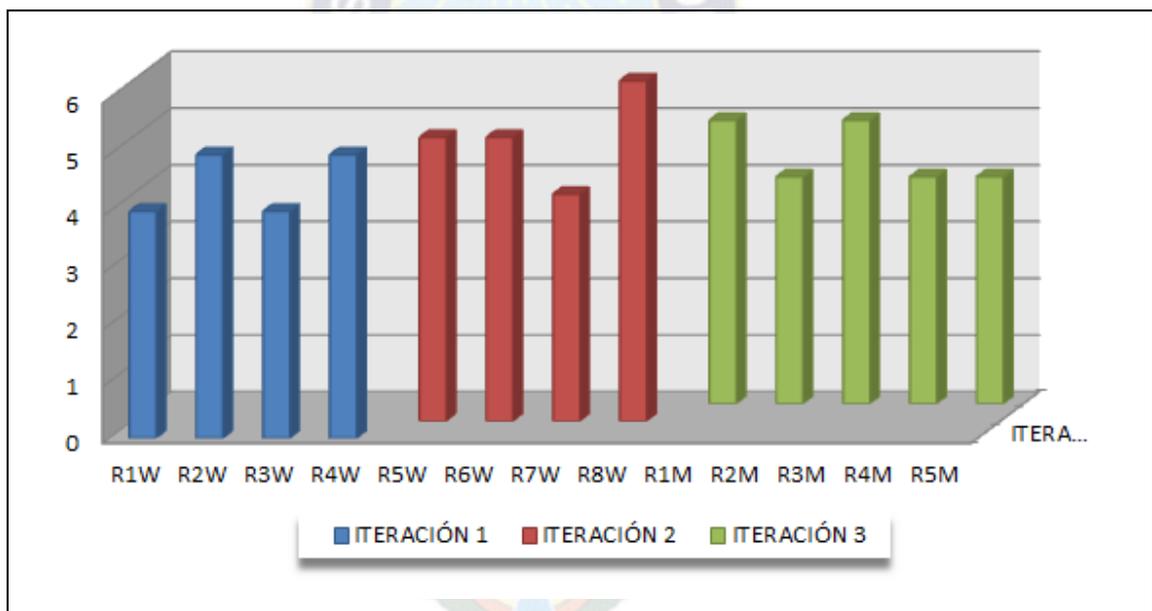
R2W representa el requerimiento 2 del desarrollo web

R1M representa el requerimiento 1 del desarrollo móvil.

Las tareas R1W hasta R5M están realizadas en las tablas 3.17 (anteriormente descritas).

El gráfico siguiente refleja en un gráfico como serán desarrolladas las tareas para el logro del objetivo final.

Figura 3.6 Desarrollo de tareas



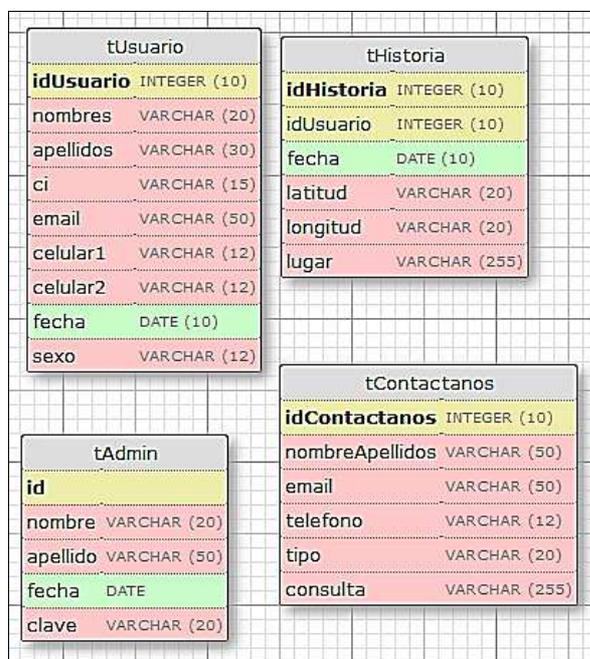
Fuente: Elaboración propia

3.4.3 Diseño de la base de datos

Las tablas identificadas para el desarrollo del sistema se presentan en base a cuatro tablas normalizadas creadas en el servidor de base de datos PhpMyAdmin.

Las tablas son: tUsuario, tContactanos, tAdmin y tHistorial.

Figura 3.7 Tablas de la Base de datos



Fuente: Elaboración propia

3.4.4 Diseño físico de la base de datos

De igual manera, las tablas siguientes detallan la información de cada estructura, considerando el nombre del campo y su tipo de dato principalmente.

- Tabla usuario.

Utilizada para registrar los datos de la persona que será sometida a un monitoreo desde la aplicación Epimovil. Guarda los datos personales.

Name	Type	Length	Decimals	Allow Null	
idusuario	int	11	0	<input type="checkbox"/>	1
nombre	char	255	0	<input checked="" type="checkbox"/>	
paterno	char	255	0	<input checked="" type="checkbox"/>	
materno	char	255	0	<input checked="" type="checkbox"/>	
email	char	255	0	<input checked="" type="checkbox"/>	
fecha	datetime	0	0	<input checked="" type="checkbox"/>	

Tabla 3.18 Estructura de la tabla usuario

Fuente: Elaboración propia

- Tabla tAdmin

La tabla tAdmin está principalmente destinada a guardar los datos de los administradores del sistema.

Esta tabla tiene dos campos que permiten el acceso al sistema para su posterior administración de las tablas de la base de datos.

Name	Type	Length	Decimals	Allow Null	
idadmin	int	11	0	<input type="checkbox"/>	 1
nombre	char	255	0	<input checked="" type="checkbox"/>	
apellido	char	255	0	<input checked="" type="checkbox"/>	
fecha	datetime	0	0	<input checked="" type="checkbox"/>	

Tabla 3.19 Estructura de la tabla tUsuario

Fuente: Elaboración propia

- Tabla tContactanos

La tabla tContactanos registra las sugerencias o quejas que tienen los usuarios que ingresan desde la página web. Esta información guarda el nombre y apellido del usuario, su correo electrónico, el número de teléfono, el tipo de consulta y una consulta donde se refleja la sugerencia o queja del usuario.

Name	Type	Length	Decimals	Allow Null	
idcontacto	int	11	0	<input type="checkbox"/>	 1
idusuario	int	11	0	<input checked="" type="checkbox"/>	
contacto	char	255	0	<input checked="" type="checkbox"/>	

Tabla 3.20 Estructura de la tabla tContactanos

Fuente: Elaboración propia

- Tabla tHistorial

Esta tabla contiene los datos del historial clínico del usuario. La información de la tabla es el aporte fundamental que el sistema provee a los diversos usuarios, en particular a los especialistas médicos presentando información mediante la plataforma web o bien mediante la aplicación móvil si el caso así lo requiere.

idHistorial	int	11	0	<input type="checkbox"/>	1
idusuario	int	11	0	<input checked="" type="checkbox"/>	
fecha	datetime	0	0	<input checked="" type="checkbox"/>	
latitud	char	255	0	<input checked="" type="checkbox"/>	
longitud	char	255	0	<input checked="" type="checkbox"/>	
audio	bigint	20	0	<input checked="" type="checkbox"/>	

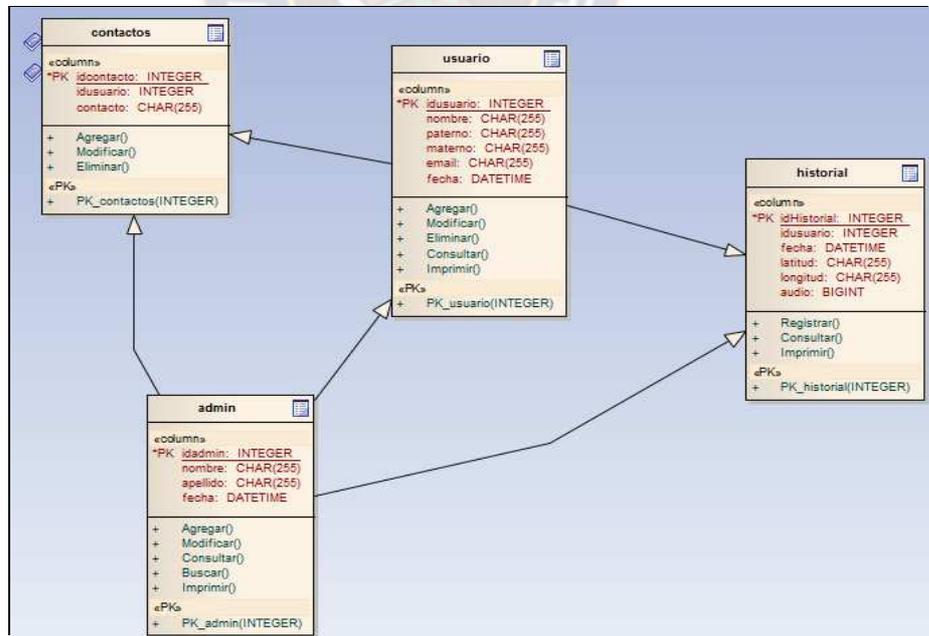
Tabla 3.21 Estructura de la tabla historial
Fuente: Elaboración propia

3.4.5 Diagrama de clases

A continuación se muestra el diagrama de clases para el sistema desarrollado presentando para cada una de las clases las operaciones asociadas que son realizadas desde la aplicación web y desde la aplicación móvil.

Las operaciones asociadas a cada una de las entidades son generalmente las de agregar, modificar, eliminar, listar, registrar, pedir ayuda. El detalle se muestra en la siguiente figura:

Figura 3.8 Diagrama de clases



Fuente: Elaboración propia

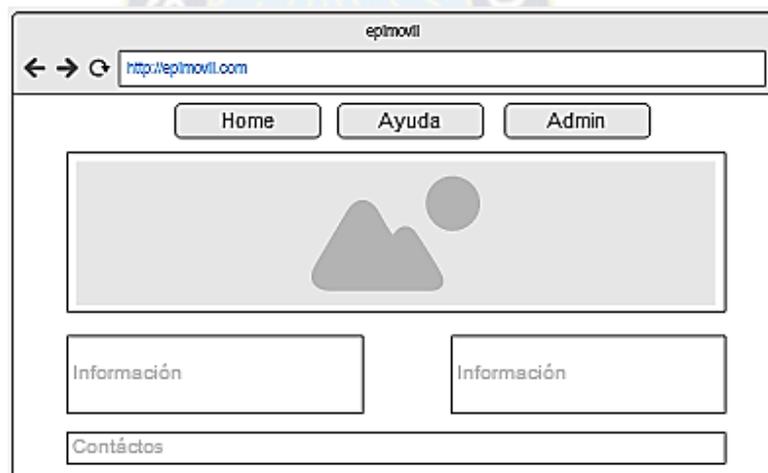
3.4.6 Diseño de pantallas web

Esta fase trata del diseño de las interfaces que serán desarrolladas en la fase de codificación. Las interfaces deben ser sencillas y fácilmente de interpretar por los usuarios y los administradores, por tanto este diseño se realizó con una herramienta opensource especialmente para elaboración de bocetos como es Moqups.

Las principales interfaces son:

Pantalla de presentación web, desde donde el usuario podrá ingresar al sistema Epimovil para que pueda acceder al portal web de administración del sistema.

Figura 3.9 Pantalla de ingreso web



Fuente: Elaboración propia

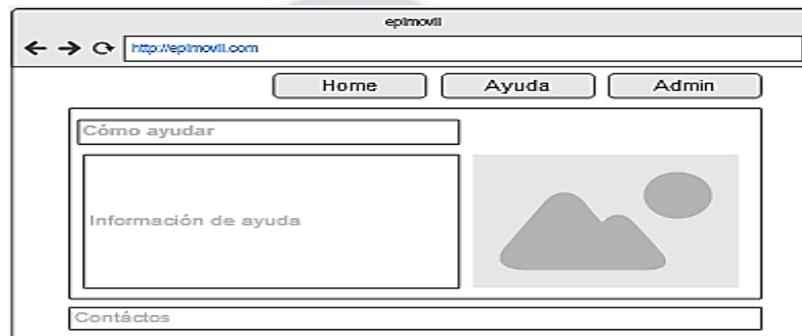
Como se observa, la pantalla de inicio solo contiene un menú y dos columnas con información relacionadas con la enfermedad que se está tratando.

En la parte inferior se presenta un botón de enlace que dirigirá a una pantalla para que el usuario pueda enviar inquietudes que serán registradas en el servidor, generalmente de sugerencias y de información que deben ser respondidas por el administrador para que haya una mayor interactividad.

Pantalla de ayuda

Esta pantalla web solo mostrará información sobre las medidas y acciones que se deben tomar cuando una persona sufre convulsiones en cualquier situación, principalmente es información de ayuda.

Figura 3.10 Pantalla de ayuda



Fuente: Elaboración propia

La opción Admin permite ingresar al módulo de administración del sistema web, principalmente el usuario debe ingresar su nombre de usuario y la contraseña asignada al mismo. La verificación de esta información con los almacenados en la base de datos permitirá ingresar o ser rechazado la solicitud de acceso.

Figura 3.11 Pantalla de ingreso web



Fuente: Elaboración propia

El administrador cuando ingresa al módulo web y siempre y cuando haya iniciado sesión, se presenta la pantalla web con las opciones que el mismo puede realizar, este diseño se presenta en la figura siguiente.

Figura 3.12 Pantalla de Administración web



Fuente: Elaboración propia

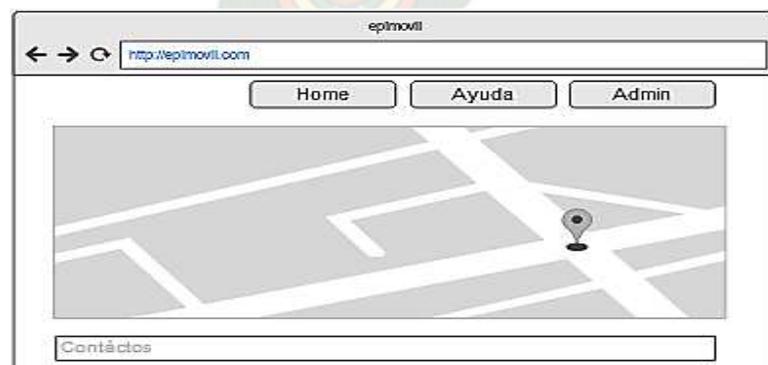
Solo se presentan cuatro opciones que presentarán los lugares geográficos donde se presentan las llamadas de ayuda de los usuarios desde su dispositivo celular.

También se presentarán estadísticas sobre los ataques sufridos por cada uno de los usuarios pacientes y un listado general.

Finalmente la última opción dará de alta a los administradores del sistema.

La primera opción tendrá el diseño que se muestra en la figura 3.13.

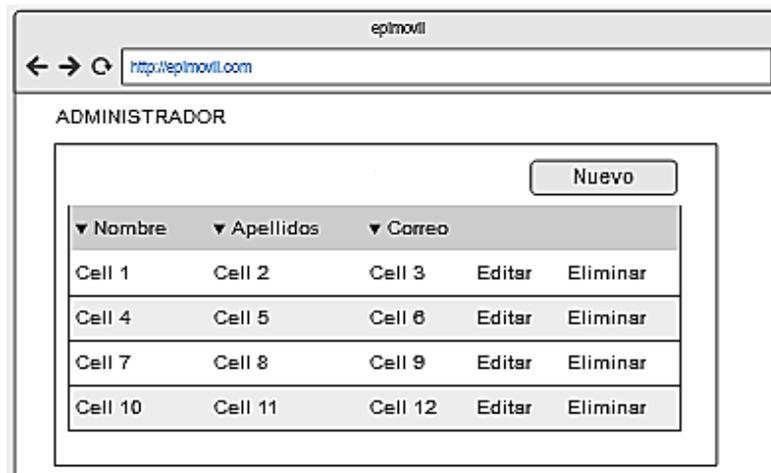
Figura 3.13 Pantalla que muestra los lugares geográficos



Fuente: Elaboración propia

Y por último se presenta el diseño de la interface correspondiente a la gestión de los usuarios autorizados que administran el sistema, presentando una lista

Figura 3.14 Pantalla CRUD de administradores



Fuente: Elaboración propia

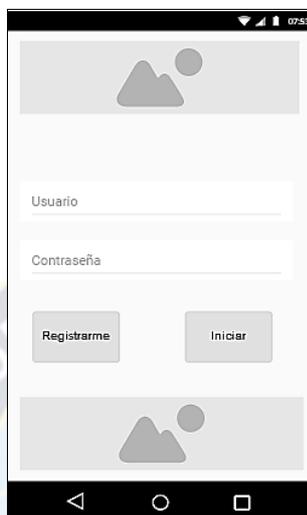
3.4.7 Diseño de pantallas aplicación móvil

La pantalla siguiente presenta la interface de ingreso cuando un usuario quiere ingresar a la aplicación. Si el usuario no está registrado, como ocurrirá la primera vez, el usuario está en obligación de registrar los datos personales desde la aplicación web y éstas serán enviadas al servidor para su almacenamiento.

La Figura siguiente refleja la pantalla de inicio de la aplicación Epimovil donde el usuario debe ingresar los datos de 'Usuario' y 'Contraseña'.

La caja de texto donde se solicita el nombre de usuario, será establecida al momento de registrarse el usuario y la contraseña será la cédula de identidad. Si los datos luego de haber sido introducidos en la interface, serán validados para su comprobación con la base de datos, si no son los correctos, el ingreso a la aplicación móvil será negado y no se inicializará la aplicación.

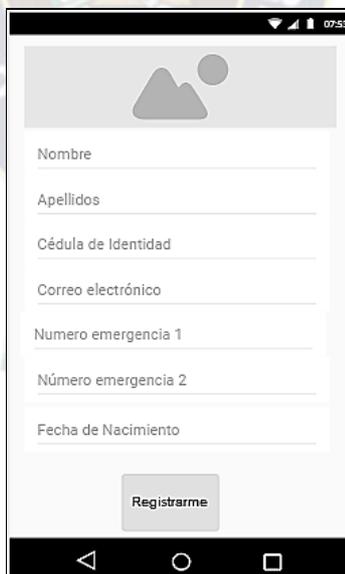
Figura 3.15 Pantalla de ingreso móvil



Fuente: Elaboración propia

En caso de no estar registrado, el usuario debe llenar datos en la siguiente pantalla

Figura 3.16 Pantalla móvil de registro de usuario



Fuente: Elaboración propia

Cuando el usuario ya inició la aplicación móvil, se presenta de manera inmediata la interface que está en constante espera 'Pedir ayuda' para que el usuario solicite ayuda y la aplicación enviará los dataos donde se presentaron los ataques de convulsiones.

Figura 3.17 Pantalla de ingreso móvil



Fuente: Elaboración propia

También la aplicación puede mostrar los lugares geográficos donde el usuario sufrió los ataques utilizando la aplicación del Google Maps para móviles.

Figura 3.18 Pantalla de ingreso móvil



Fuente: Elaboración propia

3.5 Desarrollo del sistema

El siguiente proceso que se debe realizar luego del diseño aprobado por los usuarios, es el desarrollo mismo de la aplicación web y móvil sujetos a los diseños aprobados. Entonces se presentan las siguientes interfaces desarrolladas.

3.5.1 Sistema web

Pantalla principal de ingreso al sistema web.

Figura 3.19 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

La pantalla web inicial contiene un menú con opciones principalmente de información para los diversos usuarios que ingresen.

Figura 3.20 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

La opción Admin, permite a los usuarios ingresar a la parte de administración de la aplicación web.

Pantalla de ayuda, permite mostrar información sobre cómo se debe actuar para auxiliar a personas que padezcan de convulsiones.

Ayuda a los niños.

Figura 3.21 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

Parte de la información que se muestra es:

CÓMO AYUDAR

A LOS NIÑOS

"La epilepsia se define como la presentación crónica y recurrente de fenómenos paroxísticos ocasionados por descargas neuronales desordenadas, bruscas y excesivas que se originan en el cerebro. Es uno de los trastornos neurológicos crónicos más frecuentes y eso significa un problema para la salud pública", explicó a Télam con motivo de su Día Mundial...

Pantalla de contactos

Si un usuario desea contactarse con el administrador del sistema, el sistema web ofrece una interface desde donde cualquier usuario puede solicitar información concreta sobre cualquier duda de manera directa al correo institucional del administrador.

Figura 3.22 Pantalla de envío de solicitud de ayuda

ALERTA INMEDIATA EPIMOVIL HOME AYUDA ADMIN

Direccion
Av. Villazón No 1950
La Paz - Bolivia

Telefono
(591)2451616

Correo
epimovil@gmail.com

CONTACTANOS

Nombre y apellidos(*)

Email (*)

Teléfono

Tipo de Consulta(*)
Seleccione...

Escriba su consulta

ENVIAR CONSULTA

Los campos marcados con (*) son obligatorios.

EPIMOVIL
Av. Villazón Tel.: (591) 2451616

Fuente: Elaboración propia

Pantalla de administración

Como se había indicado de manera previa, para visualizar información de la base de datos mediante la interface web, el usuario debe iniciar sesión y solo así el sistema web muestra información relacionada con las estadísticas de las convulsiones, mostrándola en una lista o bien de manera gráfica.

El usuario debe ingresar la identificación del usuario, que en este caso se refiere de manera concreta a una dirección de correo electrónico y la contraseña del mismo.

Figura 3.23 Pantalla de envío de solicitud de ayuda

ALERTA INMEDIATA HOME

INICIAR SESION

ALERTA INMEDIATA

Usuario

Contraseña

INGRESAR

Direccion
Av. Villazón No 1950
La Paz - Bolivia

Telefono
(591)2451616

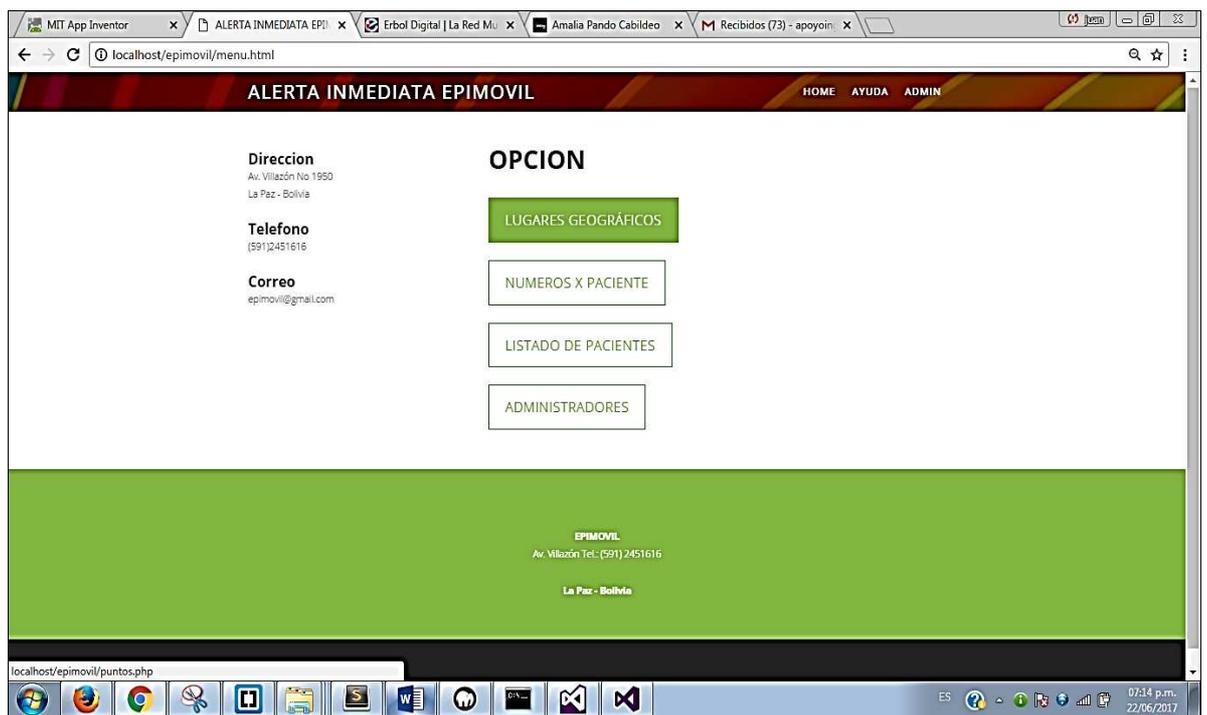
Correo
epimovil@gmail.com

Fuente: Elaboración propia

Si los datos son correctos, el sistema muestra la pantalla del administrador del sistema como se muestra en la figura 3.24.

Menú de administración

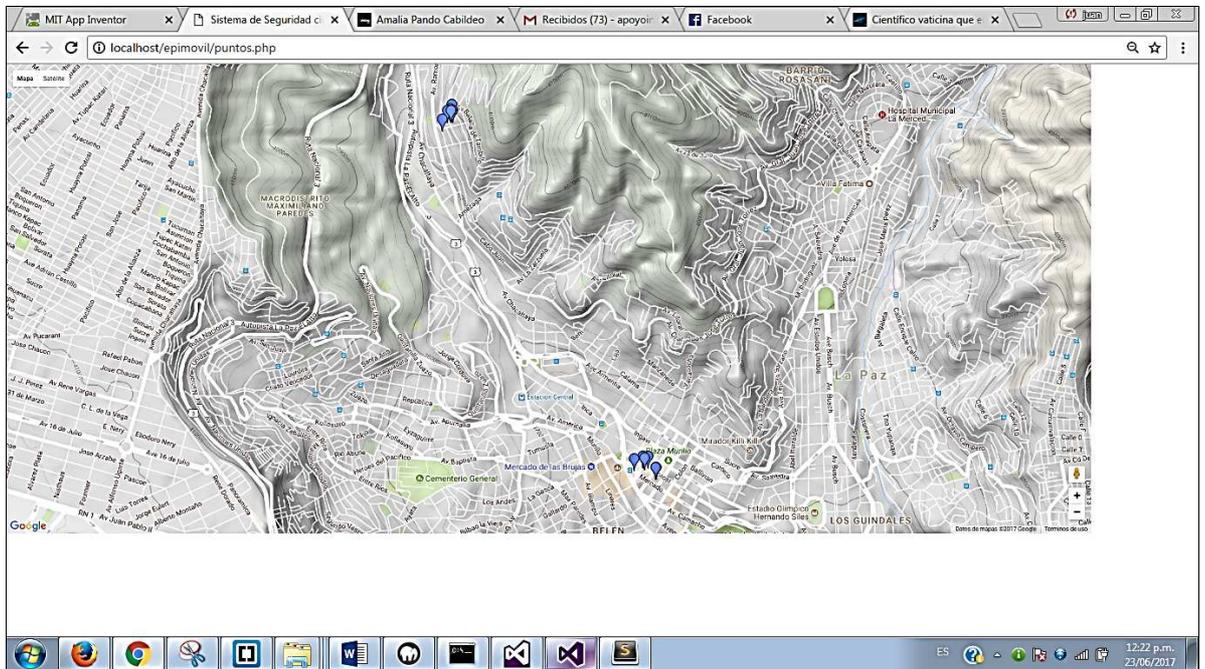
Figura 3.24 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

La primera opción, permite visualizar de forma gráfica usando el Google Maps, los puntos geográficos desde donde se han producido llamadas de ayuda de los usuarios a partir de un teléfono móvil. Cada punto geográfico es señalado por un marcador (Un globo) en el que al ser posicionado el cursor del ratón, se mostrará a manera de información la descripción del lugar, las coordenadas de latitud y longitud y la fecha en que ocurrió dicha llamada de ayuda generada por el usuario del dispositivo móvil.

Figura 3.25 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

3.5.2 Sistema móvil

Como se había establecido, el sistema web debe necesariamente trabajar con una aplicación móvil desarrollada para este propósito.

La app generada luego de elaborar la aplicación, fue subida al servidor Google Store, y desde este servidor de forma gratuita, los usuarios deben descargar la app.

Pantalla de inicio de la aplicación móvil.

Figura 3.26 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

Pantalla de registro del usuario

Figura 3.27 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

Pantalla de solicitud de ayuda inmediata

Figura 3.28 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

Figura 3.29 Pantalla de envío de solicitud de ayuda



Fuente: Elaboración propia

3.6 Estabilización

En esta fase se llegó a establecer la integración de los módulos desarrollados de forma separada en una única aplicación, realizando las siguientes tareas:

- a) La aplicación web final se subió a un servidor 000webhosting.com, estableciendo un dominio y una dirección IP determinada por el proveedor de servicios de hosting.
- b) Se creó una réplica de la base de datos en el servidor de base de datos del proveedor hosting, estableciendo el nombre de usuario y contraseña de acceso a la misma utilizando el phpMyadmin del mismo proveedor.
- c) A partir de la dirección IP asignada por el proveedor de Hosting, se modificaron en los programas desarrollados para la aplicación móvil, las solicitudes de servicios utilizando dicha IP.
- d) Se realizó las pruebas iniciales desde el dispositivo móvil con la aplicación web, se logró la correcta sincronización y no se presentaron problemas.
- e) La app generada se subió al servidor Play Store de Google, determinando que esa sea de libre uso y sin costo alguno.
- f) Finalmente los módulos web y móvil trabajan de manera sincronizada como una sola aplicación.

3.7 Pruebas

Una vez realizada la aplicación e integrada los módulos, se realizaron pruebas basadas en una encuesta considerando una muestra de la población de la ciudad de La Paz, concretamente en el Hospital de Clínicas La Paz, de forma aleatoria, a quienes se facilitó la aplicación y tras un periodo de un periodo de diez días de uso, se lograron resultados.

A continuación se presenta el cuestionario realizado para determinar la evaluación del sistema elaborado por el responsable del desarrollo de la aplicación web móvil de asistencia familiar inmediata para personas que padecen convulsiones de epilepsia.

Entrevista a usuarios

Estimado usuario Ud. ha sido seleccionado(a) para que valore el Software de Ayuda inmediata ante situaciones de Epilepsia.

¡Gracias!.	
Datos:	
Edad: _____	Fecha: _____

1. ¿Considera Ud. que la aplicación es necesario para la solicitud de ayuda inmediata y así comunicar a las personas de su confianza?

Sí _____ No _____

2 Luego de haber operado el sistema, su manejo fue

Muy fácil _____ Fácil _____ Medianamente Fácil _____ Difícil _____

3. La interface que se presenta en la aplicación móvil es:

Muy Buena _____ Buena _____ Regular _____ Mala _____

4. Después de usar la aplicación, la considera:

Muy bueno _____ Bueno _____ Regular _____ Malo _____

5. La utilidad de la aplicación y su oportuno uso es:

Muy bueno _____ Bueno _____ Regular _____ Malo _____

6¿El periodo de tiempo en que usted recibió ayuda lo considera como?

Muy bueno _____ Bueno _____ Regular _____ Malo _____

7 El seguimiento de los frecuentes ataques de epilepsia que usted sufre y su respectivo seguimiento le parece?

Muy Buena _____ Buena _____ Regular _____ Mala _____

Sugerencias.

Estas pruebas fueron contestadas por treinta usuarios, de acuerdo al periodo de evaluación.

3.7.1 Resultados de la prueba

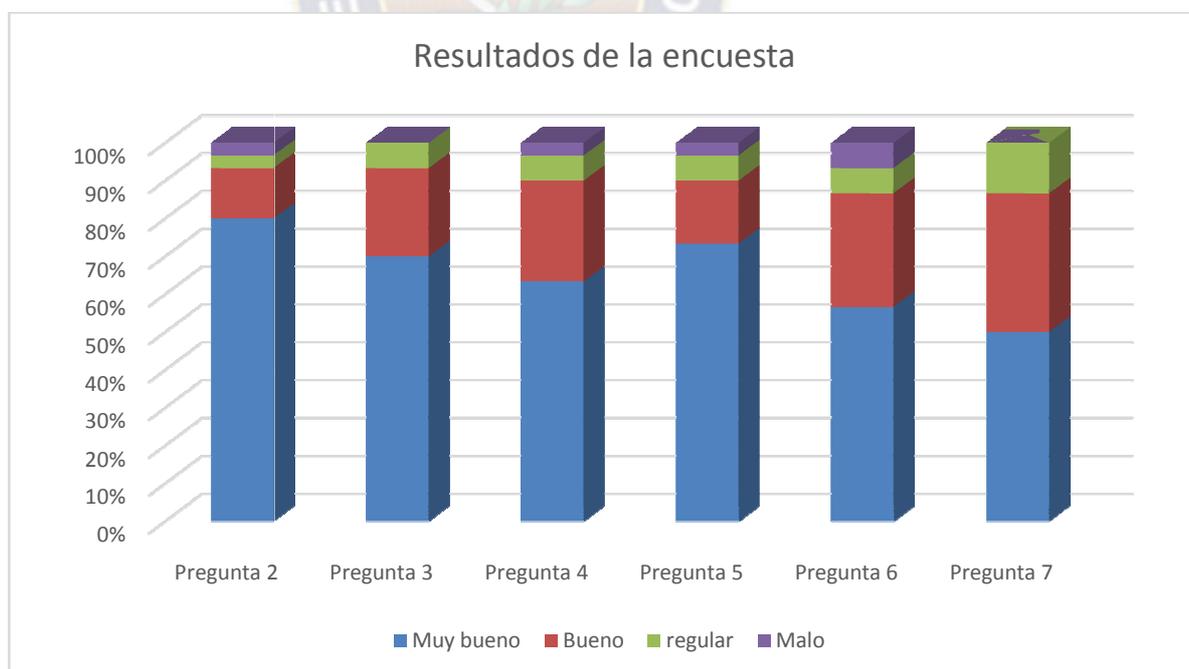
Tras la evaluación realizada por treinta personas, los resultados obtenidos se muestran en la tabla siguiente:

Calificación	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Pregunta 6	Pregunta 7
Muy bueno	24	21	19	22	17	15
Bueno	4	7	8	5	9	11
regular	1	2	2	2	2	4
Malo	1	0	1	1	2	0

Tabla 3.22 Resultados de la prueba
Fuente: Elaboración propia

De manera gráfica se muestran los resultados.

Figura 3.30 Resultados de la evaluación.



Fuente: Elaboración propia

Si se agrupan la evaluación de las siete preguntas solamente en dos categorías de respuestas como: Muy bueno / bueno y Regular / malo, los resultados reflejan claramente que la aplicación tiene plena aceptación de los usuarios que utilizaron la aplicación.

Calificación	Preg 2	Preg 3	Preg 4	Preg 5	Preg 6	Preg 7
Muy bueno / bueno	0,933	0,933	0,900	0,900	0,867	0,900
Regular / malo	0,067	0,067	0,100	0,100	0,133	0,100

Tabla 3.23 Resultados finales de evaluación

Fuente: Elaboración propia

Como se observa, los usuarios consideran a la aplicación en promedio como:

Muy bueno / bueno. 90%

Regular / malo 10%

A partir de este resultado, se puede establecer que la aplicación cumple con las expectativas de la población (por las preguntas 5 y 6 principalmente).

3.8 Calidad de Software

Para determinar la calidad del software, se utilizó las recomendaciones de la ISO 9126, considerando características del software propias a ser evacuadas.

3.8.1 Funcionalidad

Para cuantificar la funcionalidad, primero se debe ajustar los valores de punto función, basado en la siguiente tabla.

0	1	2	3	4	5
No influencia	Incidencia	Moderado	Medio	Significativo	Esencial

Tabla 3.24 Valores de Ajuste de complejidad

Fuente: Elaboración propia

La calificación del software fue elaborada por diez usuarios escogidos de manera aleatoria en inmediaciones del Hospital de Clínicas La Paz.

La tabla 3.25 muestra la cuantificación de catorce factores que se utilizan para proceder al ajuste de los puntos función, cada una de ellas evaluadas en escala del 0 (no influye) hasta el valor de 5 (esencial).

	Factor	Valor
1	Requiere el Sistema de copias de seguridad y de recuperación de datos fiables	5
2	Se requiere de comunicación de datos	5
3	Existen funciones de procesamiento distribuido	4
4	Es crítico el rendimiento	4
5	Se ejecuta el Sistema en un entorno operativo existente	5
6	Requiere el Sistema la entrada de datos de forma interactiva, mostrando múltiple pantallas u operaciones.	4
7	Se actualizan los archivos maestros de forma automática	5
8	Son complejas las entradas, salidas o peticiones de usuario	4
9	Es complejo el procesamiento interno	4
10	Es compleja la utilización del Sistema	4
11	Se ha diseñado el código para ser reutilizable	5
12	Está incluida en el diseño la instalación	4
13	Se ha diseñado para facilitar los cambios y sea de fácil uso para el usuario	4
14	Se diseñó el Sistema para soportar múltiples instalaciones en diferentes departamentos	4
	ΣFi	61

Tabla 3.25 Ajuste de complejidad del Punto Función

Fuente: Elaboración propia

De manera previa se debe estimar la cuenta total, que es un valor establecido por las características propias del software establecido por el número de entradas, número de salidas, las peticiones que realizan los usuarios, el número de archivos o tablas de la base de datos y la cantidad de interfaces desarrolladas en la aplicación.

A cada una de las características descritas, se debe calificar con un factor de peso de simple, medio y complejo y entonces la productoria y posterior suma determina la cuenta total que será utilizada para establecer el Punto Función del software.

Parámetros de medida	Cuenta	Factor de peso			Total
		Simple	Medio	Complejo	
Número de entradas del usuario	2	3	4	6	6
Número de salidas del usuario	3	4	5	7	15
Número de peticiones de usuario	4	3	4	6	16
Numero de archivos	4	7	10	15	28
Numero de interfaces	9	5	7	10	63
Cuenta Total					128

Tabla 3.26 Funcionalidad del sistema

Fuente: [Elaboración Propia]

Para calcular puntos

$$PF = \text{Cuenta Total} \times [R(t) + 0.01 \times \Sigma Fi]$$

Dónde:

Cuenta Total=Suma de todas las entradas obtenidas

0.01=Error de confiabilidad de Sistema

ΣFi = Suma de factores de complejidad

Cálculos:

$$PF = 128 * (0.65 + 0.01 * 61) = 161.28$$

Por otra parte, si consideramos la evaluación de los catorce factores de la tabla 3.25, se tiene un factor máximo de 70. Entonces el PF máximo será:

$$PF \text{ Máximo} = 128 * (0.65 + 0.01 * 70) = 172.8$$

$$\text{Funcionalidad} = \frac{\text{punto función del software}}{\text{punto función máximo}} * 100$$

$$= (161.28 / 172.8) * 100$$

$$= 93.3$$

Entonces la funcionalidad del Sistema es de 93%

3.8.2 Fiabilidad

La fiabilidad se refiere al punto en que se puede esperar que el programa lleve a cabo su función predefinida con la exactitud requerida y está dado por:

$$Fiabilidad = 1 - \frac{\# \text{ de errores}}{\# \text{ de líneas de código}}$$

Duración	Inicio	Fin	Módulos
1	11 May 2017	11 May 2017	Módulo web Informaciones Ayuda Registro de Administradores Módulo de administración Listado de usuarios Registro de ayudas. Consultas Reporte gráfico basado en Google Maps
2	7 Jun 2017	8 Jun 2017	Módulo de la aplicación móvil Registro de usuarios Registro de números de contacto Ayuda Reporte gráfico basado en Google Maps
10	12 Jun 2017	22 Jun 2017	Módulo web Módulo de la aplicación móvil

Tabla 3.27 Resultados de evaluación de ejecución del sistema

Fuente: Elaboración propia

Se realizaron tres pruebas de acuerdo a lo planificado, una prueba a la parte de la aplicación web; otra prueba referida a la aplicación móvil desarrollada como complemento para enviar datos generados por los usuarios y una prueba final considerando el funcionamiento sincronizado de las aplicaciones web y móvil como una sola aplicación.

Las pruebas fueron realizadas un día de 8 horas, 2 días y finalmente diez días, en cada periodo de evaluación se considera transacciones y los resultados registrados se muestran en la tabla siguiente.

Tiempo de servicio a	Transacciones realizadas b	Fallas encontradas c	Probabilidad fallo bajo demanda $d = c/b$	Tiempo medio entre fallos $e = a/c$
8 hrs	70	52	0.074	0.15 hrs.
16 hrs	54	34	0.629	0.47 hrs.
160 hrs.	1500	22	0.014	7.2 hrs.

Tabla 3.28 Resultados de evaluación de ejecución del sistema
Fuente: Elaboración propia

El promedio de errores encontrados es de $108 / 3 = 36$ errores en las tres pruebas de ejecución del software.

$$\text{Fiabilidad} = 1 - (36 \text{ errores} / 3260 \text{ líneas de código})$$

$$\text{Fiabilidad} = 0.988 * 100$$

$$\text{Fiabilidad} = 98\% (\text{Significa que en } 98\% \text{ el sistema es fiable})$$

3.8.3 Facilidad de mantenimiento

Es el esfuerzo necesario para localizar y arreglar un error en el programa y está dada por la siguiente ecuación matemática:

$$= 1 - 0.1 * (\text{número medio de días - hombre por corrección})$$

$$\text{Facilidad de mantenimiento} = 1 - 0.1 (2 - 1 \text{ persona por corrección})$$

$$\text{Facilidad de mantenimiento} = 0.9 * 100$$

$$\text{Facilidad de mantenimiento} = 90\% (\text{Significa que en un } 90\% \text{ es fácil de mantener})$$

3.8.4 Eficiencia

Del grupo de usuarios inicialmente treinta, un grupo de diez personas fueron seleccionadas para evaluar aspectos inherentes al software.

Para evaluar este factor de calidad, se consideraron los atributos correspondientes a la característica de eficiencia compuesto por la comprensibilidad del sistema, mecanismos de

ayuda y retroalimentación, aspectos de la interfaz, aspectos de exploración y los errores que se pudieron haber presentado.

Característica	Adm1	Adm2	Usr1	Usr2	Usr3	Usr4	Usr5	Usr6	Usr7	Usr8
Comprensibilidad del sistema	90	95	95	80	95	95	93	92	89	95
Mecanismos de ayuda y retroalimentación	93	96	93	91	93	90	94	96	92	92
Aspectos de la interfaz	92	95	95	90	85	86	94	90	95	92
Aspectos de exploración	94	90	90	95	92	90	93	92	93	90
Errores	10	5	10	8	12	6	11	8	9	5

Tabla 3.29 Resultados de evaluación según encuesta a los usuarios

Fuente: Elaboración propia

La evaluación se considera en una escala del 1 hasta 100 puntos, siendo 100 puntos considerado como mejor calificación. En la tabla 3.29, como se observa, se tienen diez evaluaciones relacionados a los aspectos referidos y los resultados promedios se muestran en la tabla 3.30.

CARACTERÍSTICA	TOTAL
Comprensibilidad del sistema	91.9
Mecanismos de ayuda y retroalimentación	93.0
Aspectos de la interfaz	91.4
Aspectos de exploración	91.9
Errores	8.4

Tabla 3.30 Resultados de evaluación de eficiencia del sistema

Fuente: Elaboración propia

Los resultados obtenidos en base a un cuestionario resuelto por los usuarios, se consideró un total de 10 encuestas.

A partir de los datos logrados y calculando el promedio de estos resultados, la eficiencia se determina como:

$$\text{Eficiencia} = \text{promedio } (95, 93, 91.4, 91.9, (100-8.4))$$

$$\text{Eficiencia} = 92.8\%$$

Este resultado indica que el Sistema elaborado logra un servicio con una eficiencia del 93%, que se interpreta como un rendimiento satisfactorio del software.

3.8.5 Flexibilidad

La flexibilidad es el coste de modificación del producto cuando cambian sus especificaciones, y puede ser calculada a partir de la siguiente fórmula:

$$1 - 0.05 \text{ (número medio de días – hombre por cambio)}$$

$$\text{Flexibilidad} = 1 - 0.05 (2 - 1)$$

$$\text{Flexibilidad} = 0.95 * 100$$

$$\text{Flexibilidad} = 95\%$$

Entonces el sistema tiene una flexibilidad del 95%

3.9 Estimación de costos

Este apartado trata sobre la estimación del costo de desarrollo del sistema. Para su determinación, ésta se determina utilizando el modelo COCOMO y COCOMO II.

En primera instancia según el modelo COCOMO.

- MODELO COCOMO

ESFUERZO

Las líneas de código se determinan en base a la relación lenguaje de programación – puntos de función, para el caso del presente trabajo como se utilizó un lenguaje orientado a objetos, entonces se tiene la relación de 1 PF = 32 LDC.

Ecuación 1 Esfuerzo de desarrollo estimado

$$E = 2.4 * \left[\frac{(161 * 32)}{1000} \right]^{1.05}$$

$$E = 13.4 \text{ personas por mes}$$

TIEMPO DE DESARROLLO

Para el cálculo del tiempo se utiliza nuevamente la expresión de COCOMO para la determinación del tiempo estimado inicial del proyecto.

Ecuación 2 Tiempo de desarrollo estimado

$$D = 2.5 * [13.2]^{0.38}$$

$$D = 6.6 \text{ meses}$$

Considerando el sueldo mínimo en el Estado Plurinacional de Bolivia, a fecha Julio de 2017, un empleado promedio tiene un sueldo de Bs. 2.000.- aproximadamente, por lo que se determina que el costo de desarrollo es:

$$\text{COSTO ESTIMADO} = 6.6 * 2.000 = 13200 \text{ Bs.}$$

- SEGÚN COCOMO II utilizando software COCOMO II ofrecido de manera gratuita como herramienta case por CSSE.

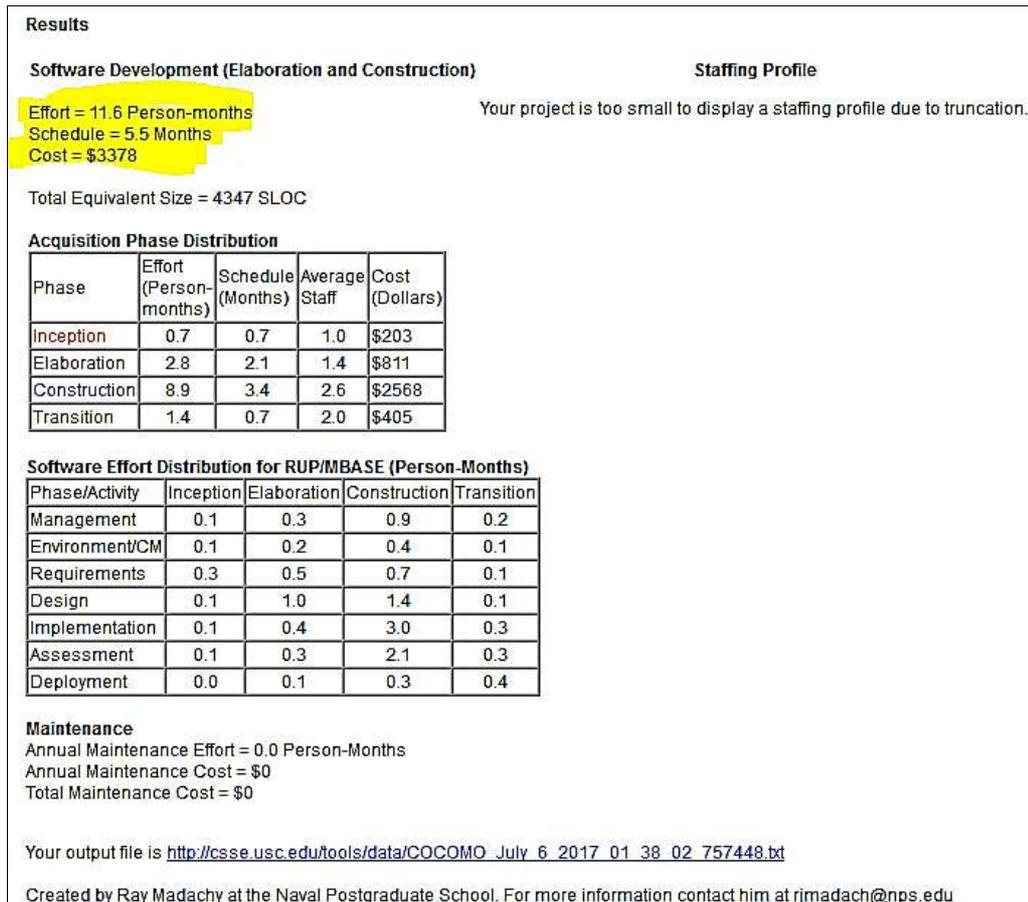
Por otra parte, utilizando el software CSSE, se realiza el cálculo según las características del sistema.[14]

Figura 3.31 Datos de entrada COCOMO II

Fuente: Elaboración propia

Los resultados generados se presentan en la siguiente captura de pantalla:

Figura 3.32 Datos de salida COCOMO II



Fuente: Elaboración propia

Nótese en el gráfico que los resultados resaltados refieren un tiempo de desarrollo de 5.5 meses y un trabajo de 11 personas mes.

Al haber establecido que el sueldo mínimo es de 290 dólares americanos (equivalente a Bs. 2000), entonces el costo estimado de desarrollo calculado por el software es de 3378 dólares americanos.

Finalmente como conclusión se tiene:

Método	Costo estimado	
	Bs.	Dólares EE.UU.
COCOMO	13.200	1.893
COCOMO II	23.544	3.378

Tabla 3.31 Costo estimado de desarrollo de software
Fuente: Elaboración propia

Al ser el método COCOMO II más detallado y tomar más parámetros para la estimación final, se concluye que este costo debe ser considerado como un costo máximo referencial.





CAPITULO IV CONCLUSIONES Y RECOMENDACIONES

4 CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

Con el desarrollo e implementación de la aplicación móvil de ayuda inmediata para personas que sufren de ataques de epilepsia, se llegaron a las siguientes conclusiones.

4.1.1 De la hipótesis

La aplicación móvil georeferenciada coadyuva en el auxilio inmediato a personas que padecen de epilepsia logrando mostrar en un mapa la ubicación exacta donde se encuentra y reduciendo el tiempo de asistencia.

Según 3.7.1, los usuarios que han evaluado la aplicación móvil de ayuda inmediata y que han sufrido ataques de epilepsia, consideran que la aplicación les ha sido útil en 90%.

4.1.2 De los objetivos

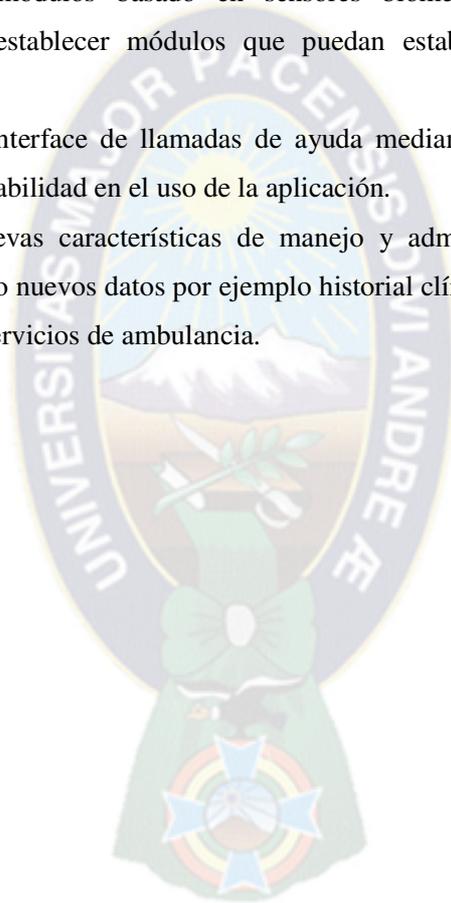
- Se logró desarrollar una aplicación web - móvil georeferenciada para alerta inmediata destinada a prestar ayuda oportuna a las personas que padecen convulsiones de epilepsia mediante mensajes y alertas enviadas a los contactos familiares o personas cercanas al entorno del afectado logrando reducir el tiempo de asistencia médica. La aplicación móvil envía de manera alterna mensajes mediante la línea telefónica como también mediante llamadas por la aplicación whatsapp, en particular éste último medio es el que está habilitado por razones de economía.
- La metodología ágil Mobile-D utilizada permitió orientar el desarrollo de la aplicación cumpliendo de manera estricta los objetivos la estimación de tiempo de todos los módulos involucrados en el desarrollo de los sistemas web y móvil.
- Se dispone de un medio de alerta y comunicación dirigido al grupo familiar cuando una persona sufre una convulsión o ataque.
- Se dispone de un registro sobre la frecuencia y lugar donde la persona sufre estas convulsiones de epilepsia almacenadas en una tabla en el servidor de base de datos. Entonces las consultas pueden ser efectuadas por persona, por periodos y ser reflejadas en un mapa utilizando librerías de Google Maps.

4.2 Recomendaciones

Según la investigación realizada en la elaboración de este trabajo, se recomienda seguir los siguientes trabajos futuros.

Trabajos futuros.

- Incorporar módulos basados en sensores biométricos de detección de presión sanguínea, establecer módulos que puedan establecer las solicitudes de auxilio inmediatas.
- Establecer interface de llamadas de ayuda mediante análisis de voz y así dar una mayor confiabilidad en el uso de la aplicación.
- Agregar nuevas características de manejo y administración de la base de datos, incorporando nuevos datos por ejemplo historial clínico compartidas con instituciones médicas y servicios de ambulancia.



5 Bibliografía

- [1] M. Velarde, «En Bolivia 3 de cada 100 personas confrontan la crisis de la epilepsia,» *El Diario*, 29 Abril 2012.
- [2] F. Chávez, «Un 83% de los bolivianos tiene un teléfono celular,» *Página 7*, 10 Noviembre 2013.
- [3] dmedicina, «dmedicina.com,» 16 Septiembre 2015. [En línea]. Available: <http://www.dmedicina.com/enfermedades/neurologicas/epilepsia.html>.
- [4] B. Blanchard, *Ingeniería de Sistemas*, Madrid: Isdefe, 1995.
- [5] R. Pressman, *Ingeniería de Software, un enfoque práctico*, México: McGraw Hill, 2010.
- [6] V. Viera, *Computación Móvil. Principios y técnicas*, 2010.
- [7] R. Ramírez, *Métodos para el desarrollo de aplicaciones móviles*, Universidad de Catalunya, 2010.
- [8] E. Rodríguez, «La Geolocalización, Coordenadas hacia el éxito,» Universidad Salamanca, 2010.
- [9] J. Ball, «Angry Birds and 'leaky' phone apps targeted by NSA and GCHQ for user data,» *World news, the Guardian*, 28 Enero 2014.
- [10] Google_Maps, «Google Maps APIs,» 01 Junio 2016. [En línea]. Available: <https://enterprise.google.com/intl/es-419/maps/products/mapsapi.html>. [Último acceso: 13 Junio 2017].

- [11] D. Anderson, «CakePHP 1.3 Cookbook,» Manual de CakePHP, 2012. [En línea]. Available: <https://book.cakephp.org/1.3/es/index.html>. [Último acceso: 10 Junio 2017].
- [12] ISO, *ISO/IEC 9126-1:2001*, 2011.
- [13] R. Rodríguez, Sistema de Control de Personal y Planillas de pago, La Paz: UMSA, 2011.
- [14] P. Nava, Sistema de seguimiento y control a colegiados departamentales fundamentada en Multiagentes, La Paz: Universidad de Aquino Bolivia, 2014.
- [15] I. Sommerville, Ingeniería del software, Pearson Educación, 2005.







ANEXOS

La Paz 8 de Junio de 2017

Señor

Lic. Edgar Palmiro Clavijo Cardenas

DIRECTOR

CARRERA DE INFORMATICA

FAC. CIENCIAS PURAS Y NATURALES

CARRERA DE INFORMATICA

Presente

Ref. AVAL PARA LA DEFENSA DE TESIS DE GRADO

De mi mayor consideración:

Mediante la presente, me dirijo ante su Autoridad, en calidad de Tutor Metodológico para informar que luego de haber realizado el seguimiento de la Tesis de Grado titulada: “**APLICACIÓN MÒVIL DE ASISTENCIA FAMILIAR INMEDIATA PARA PERSONAS QUE PADECEN CONVULSIONES DE EPILEPSIA**”, presentado por el Univ. Ronald Hernan Cruz Arias con C.I. 4331444 LP, para optar al título de Licenciatura en Informática con mención Ingeniería de Sistemas Informáticos.

En este sentido, presento mi **conformidad y aval** respectivo para la defensa pública del Proyecto de Tesis de acuerdo a Reglamento vigente en la Universidad Mayor de San Andrés.

Sin otro particular, me suscribo con las atenciones más distinguidas.

Lic. Freddy Miguel Toledo Paz

TUTOR METODOLOGICO

c.c. Arch

La Paz 8 de Junio de 2017

Señor

Lic. Freddy Miguel Toledo Paz

TUTOR METODOLOGICO

Presente

Ref. CONFORMIDAD Y AVAL DE TESIS DE GRADO

De mi mayor consideración:

Tengo a bien dirigirme ante su persona, para darle a conocer que luego de efectuar el seguimiento a la estructura y contenido de la Tesis de Grado titulada: **“APLICACIÓN MÒVIL DE ASISTENCIA FAMILIAR INMEDIATA PARA PERSONAS QUE PADECEN CONVULSIONES DE EPILEPSIA”**, elaborado por el Univ. Ronald Hernan Cruz Arias con C.I. 4331444 LP, en calidad de **Asesor** expreso mi conformidad con el contenido y la forma de trabajo, dando **mi Aval** para que el postulante pueda realizar la defensa de Tesis de Grado para optar al título de Licenciatura en Informática con mención Ingeniería de Sistemas Informáticos, de acuerdo a normas y reglamento vigentes.

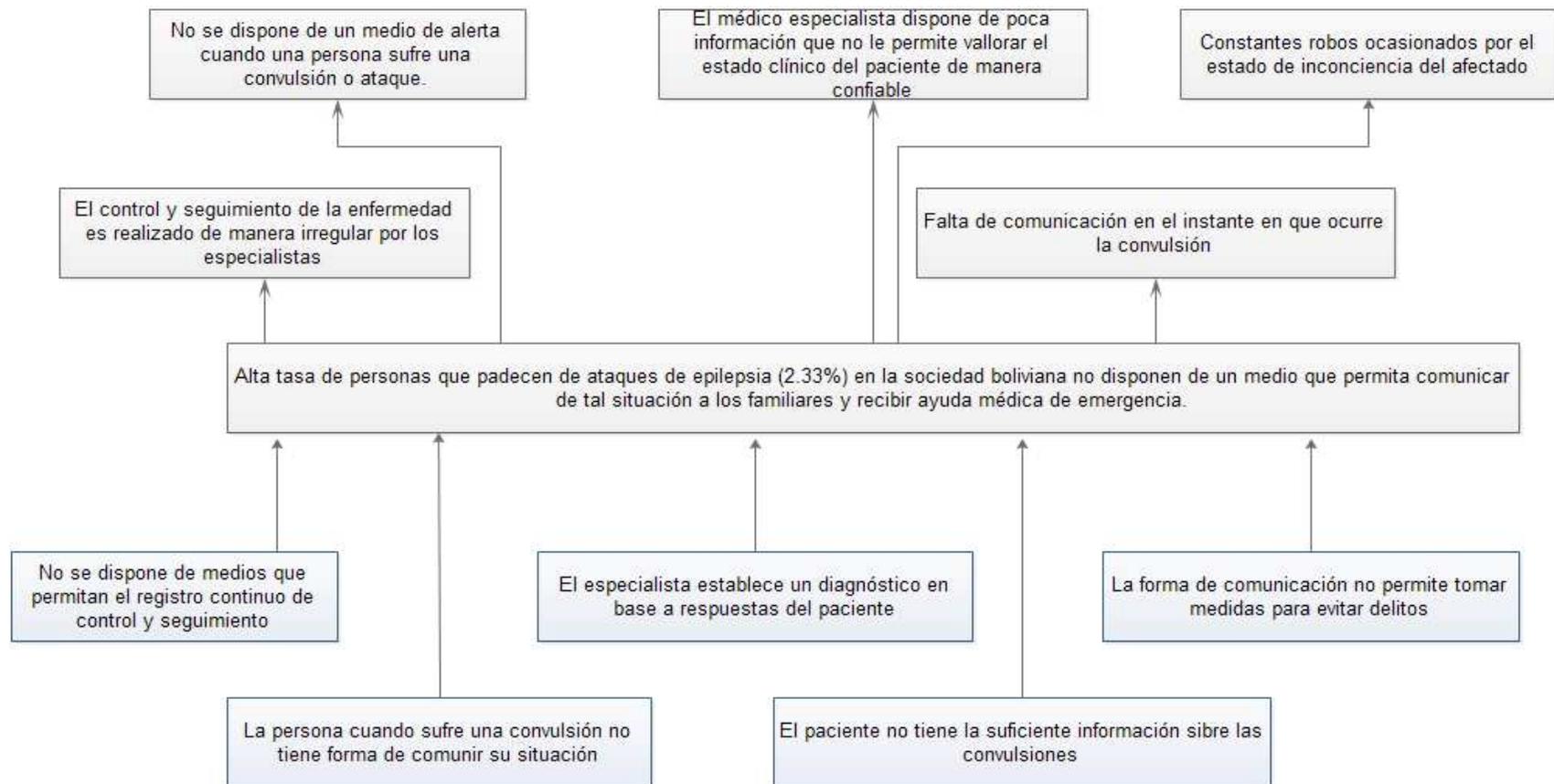
Sin otro particular, me suscribo de su persona con las consideraciones más distinguidas.

M. Sc. Carlos Mullisaca Choque

DOCENTE ASESOR

c.c. Arch

ANEXO A. ÁRBOL DE PROBLEMAS



ANEXO B. ÁRBOL DE OBJETIVOS

