



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E SCIENTIA HOMINIS SALUS "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DESARROLLO DE UN PROTOTIPO DE UN SISTEMA DE APLICACIÓN MÓVIL DE GESTIÓN MÉDICA PARA EL ÁREA DE TRAUMATOLOGÍA EN LA CLÍNICA “CLUB DE LEONES QUITO”

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN “ELECTRÓNICA Y REDES DE INFORMACIÓN”**

ALEXIS DAVID CHANGO GUANO

alexis.chango@epn.edu.ec

DIRECTOR: PhD. ANA MARÍA ZAMBRANO VIZUETE

ana.zambrano@epn.edu.ec

Quito, noviembre 2018

AVAL

Certifico que el presente trabajo fue desarrollado por Alexis David Chango Guano, bajo mi supervisión.

PhD. ANA ZAMBRANO
DIRECTORA DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORIA

Yo Alexis David Chango Guano, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

ALEXIS DAVID CHANGO GUANO

DEDICATORIA

Dedico este trabajo a mis padres Edwin y Blanca, que han sido un apoyo incondicional a lo largo de toda mi vida, siendo una fuente de guía y apoyo, aprovecho este espacio para decirles que los amo y que agradezco a Dios por darme la bendición de tenerlos y sé que se sienten orgullosos de que cumpla esta meta en mi vida. A mis hermanos Fabián, Xavier y Esteban que de cierta manera han apoyado a lo largo de este objetivo. A mi hijo Josué que viene en camino, eres mi mayor inspiración en la vida

David

AGRADECIMIENTO

Agradezco en primer lugar a Dios, que me ha dado muchas bendiciones a lo largo de estos años de estudio, y se que sin él no hubiese podido culminar con el mayor de los éxitos esta meta en mi vida.

Agradezco a mis padres Edwin y Blanca que han sido una fuente inagotable de consejos y mucha sabiduría, de tenerme paciencia durante estos años de formación profesional.

Agradezco a la Escuela Politécnica Nacional, a esta gran institución que ha formado grandes profesionales durante muchos años.

Agradezco a todos mis amigos y compañeros de esta etapa universitaria que hemos compartido gratos momentos, que llevaré por siempre en mi corazón.

Agradezco a Evelyn una persona muy especial que me ha brindado todo su apoyo en momentos difíciles y ha hecho que siempre saque la mejor versión de mí.

Agradezco a la Dra. Ana Zambrano que ha sido mi guía en la elaboración de este proyecto, que con sus sabios y acertados consejos se ha podido desarrollar con el mayor de los éxitos.

Y finalmente agradezco a todos los profesores de la Escuela Politécnica Nacional que con sus conocimientos han hecho que me forme como profesional y culminen mis estudios en esta gran institución.

David

ÍNDICE DE CONTENIDO

AVAL.....	I
DECLARACIÓN DE AUTORIA	II
DEDICATORIA	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
ÍNDICE DE FIGURAS	VII
ÍNDICE DE TABLAS.....	X
ÍNDICE DE CÓDIGOS	XI
RESUMEN.....	XII
ABSTRACT.....	XIII
1. INTRODUCCIÓN.....	1
1.1 Objetivos	1
1.2 Alcance	1
1.2.1 Administración del Médico	3
1.2.2 Administración del Paciente	3
1.3 Marco Teórico	4
1.3.1 Estado del Arte.....	5
1.3.2 Smartphones y Aplicaciones Móviles	10
1.3.3 Cloud Computing	12
1.3.4 Descripción de Herramientas y Tecnologías Utilizadas.....	25
1.3.5 Metodología Ágil de Desarrollo Kanban	28
2. METODOLOGÍA.....	31
2.1 Diseño	31
2.1.1 Tablero de Actividades Kanban.....	31
2.1.2 Análisis de Requerimientos	32
2.1.3 Selección del Proveedor de Cloud Computing	33
2.1.4 Diseño de la Capa de Datos.....	36
2.1.5 Diseño de la Capa de Negocio	37
2.1.6 Diseño de la Capa de Presentación	44
2.2 Implementación.....	49
2.2.1 Tablero de Actividades Kanban.....	49
2.2.2 Instalación de Herramientas Necesarias.....	50
2.2.3 Codificación de la Capa de Base de Datos	56

2.2.4 Codificación de la Capa de Negocio	57
2.2.5 Codificación de la Capa de Presentación	61
3. RESULTADOS Y DISCUSIÓN	70
3.1 Tablero de Actividades Kanban	70
3.2 Pruebas de Validación de Requerimientos Funcionales	70
3.2.1 Pruebas de Validación del Subsistema 01: Creación de Perfil.....	71
3.2.2 Pruebas de Validación del Subsistema 02: Gestión de Información Personal	72
3.2.3 Pruebas de Validación del Subsistema 03: Gestión de Pacientes	73
3.2.4 Pruebas de Validación del Subsistema 04: Gestión de atención al paciente.....	76
3.2.5 Pruebas de Validación del Subsistema 05: Gestión de historiales clínicos	78
3.2.6 Pruebas de Validación del Subsistema 06: Gestión de actividades del paciente.....	81
3.3 Pruebas de Validación de Requerimientos No Funcionales	84
3.3.1 RNF01: Presentación.....	89
3.3.2 RNF02: Disponibilidad	90
3.3.3 RNF03: Integridad.....	90
3.3.4 RNF04: Tiempo de Respuesta	91
3.3.5 RNF05: Escalabilidad	91
3.3.6 RNF06: Capacidad.....	91
3.4 Tablero Final de Kanban	93
4. CONCLUSIONES Y RECOMENDACIONES	95
4.1 Conclusiones	95
4.2 Recomendaciones	96
5. REFERENCIAS BIBLIOGRÁFICAS	97
6. ANEXOS	102
ORDEN DE EMPASTADO	103

ÍNDICE DE FIGURAS

Figura 1.1. Esquema del Proyecto de Titulación.	2
Figura 1.2. Esquema de la Aplicación de Procedimientos Quirúrgicos.	6
Figura 1.3. Diagrama de flujo del funcionamiento de la aplicación de Procedimientos Quirúrgicos.	7
Figura 1.4. Esquema de aplicación ICUC.	8
Figura 1.5. Uso de la aplicación en un tratamiento de rehabilitación.	9
Figura 1.6. Tipo de aplicaciones médicas y porcentaje de utilización.	12
Figura 1.7. Ejemplos de tipo de servicio IAAS.	13
Figura 1.8. Ejemplos de tipo de servicio PAAS.	14
Figura 1.9. Ejemplo de tipo de servicio SAAS.	14
Figura 1.10. Cloud Público.	16
Figura 1.11. Cloud Privado.	17
Figura 1.12. Cloud Híbrido.	17
Figura 1.13. Cloud Comunitario.	18
Figura 1.14. Arquitectura de un sistema utilizando un servicio web en la nube.	25
Figura 1.15. Archivos de un proyecto en la vista de Android.	27
Figura 1.16. Ventana principal de Android Studio.	28
Figura 1.17. Ejemplo de un tablero con tarjetas de trabajo Kanban.	29
Figura 2.1. Tablero de Actividades Kanban del Diseño.	31
Figura 2.2. Cuadrante Mágico de Gartner.	34
Figura 2.3. Diagrama Entidad-Relación de la capa de base de datos.	37
Figura 2.4. Actores que interactúan con el sistema.	37
Figura 2.5. Subsistemas que conforman el Sistema de Gestión Médica.	38
Figura 2.6. Caso de uso registro de médico.	39
Figura 2.7. Caso de uso de gestión de información personal.	40
Figura 2.8. Caso de uso de gestión de pacientes.	40
Figura 2.9. Caso de uso de gestión de atención de pacientes.	41
Figura 2.10. Casos de uso de gestión de historiales.	41
Figura 2.11. Caso de uso de gestión de actividades del paciente.	42
Figura 2.12. Diagrama de Clases.	43
Figura 2.13. Arquitectura del Servicio Web.	44
Figura 2.14. Interfaces de registro de médicos.	45
Figura 2.15. Pantallas de visualización del Perfil del usuario médico.	45
Figura 2.16. Interfaces de registro de pacientes.	46
Figura 2.17. Interfaces de la gestión de información del paciente.	46
Figura 2.18. Interfaz de sugerencia de cita médica.	47
Figura 2.19. Interfaz de visualización de sugerencia de citas médicas.	47
Figura 2.20. Interfaces de visualización para mostrar el historial completo.	48
Figura 2.21. Interfaces de acceso al sistema usuario Paciente.	48
Figura 2.22. Diseño de interfaces del usuario paciente.	49
Figura 2.23. Tablero de Actividades Kanban de la Implementación.	50
Figura 2.24. Arquitectura del servidor.	51
Figura 2.25. Selección del servicio para la instalación de una nueva máquina virtual.	51
Figura 2.26. Botón para crear una instancia de AWS.	52

Figura 2.27. Opciones de Imágenes de AWS.....	52
Figura 2.28. Características del tipo de instancia escogida.	52
Figura 2.29. Resumen de las características del tipo de Servidor escogido.	53
Figura 2.30. Interfaz lista para conectar al servidor remoto en la nube AWS.....	53
Figura 2.31. Interfaz del Servidor en la Nube.	53
Figura 2.32. Descarga de WampServer de 64 bits.....	54
Figura 2.33. Servicios activos de WampServer.	54
Figura 2.34. Instalación de NetBeans.	55
Figura 2.35. Aplicaciones de NetBeans.	55
Figura 2.36. Creación del directorio para los servicios Web.	56
Figura 2.37. Interfaces del sistema de Gestión Médica en Android.....	61
Figura 2.38. Pantalla de registro usuario Médico.	62
Figura 2.39. Pantalla de registro paciente nuevo.	63
Figura 2.40. Pantallas para la Gestión de Historiales Clínicos.....	64
Figura 2.41. Pantallas de Cargar Exámenes y Mostrar Exámenes.	64
Figura 2.42. Pantalla de mostrar historial y creación de diagnósticos.	65
Figura 2.43. Pantallas de Crear Diagnóstico y Mostrar Historial Completo	65
Figura 2.44. Pantalla Eliminar Paciente.	66
Figura 2.45. Pantallas para la sugerencia de citas médicas.	66
Figura 2.46. Pantallas para la visualización de citas médicas.	67
Figura 2.47. Pantalla de visualización Mi Perfil	67
Figura 2.48. Pantalla Principal Usuario Paciente y visualización de información personal.....	68
Figura 2.49. Pantallas de visualización de sugerencia.....	68
Figura 2.50. Pantallas de sugerencias o quejas.....	69
Figura 2.51. Pantallas para cargar Exámenes Médicos.....	69
Figura 3.1. Tablero Kanban para la etapa de resultados y pruebas.	70
Figura 3.2. Mensaje de validación de campos.....	71
Figura 3.3. Mensaje de confirmación de registro de nuevo médico.....	71
Figura 3.4. Acceso usuario médico.....	72
Figura 3.5. Mensaje de validación de datos de usuario.	72
Figura 3.6. Visualización y modificación de datos de usuario médico.	73
Figura 3.7. Mensaje de error al momento de actualizar.	73
Figura 3.8. Mensaje de paciente registrado exitosamente.....	74
Figura 3.9. Mensaje de paciente ya creado.....	74
Figura 3.10. Actualización información paciente.	75
Figura 3.11. Prueba de eliminación de pacientes.....	75
Figura 3.12. Buscador de pacientes.	75
Figura 3.13. Sugerencia de visitas médicas y agenda.	76
Figura 3.14. Mensaje de notificación de visitas médicas.	77
Figura 3.15. Sugerencias y quejas de pacientes.....	77
Figura 3.16. Mensaje de notificación de sugerencias.....	77
Figura 3.17. Carga de exámenes médicos.	78
Figura 3.18. Eliminación de exámenes médicos.	79
Figura 3.19. Historial completo y creación de diagnósticos.	79
Figura 3.20. Descripción del estado de salud del paciente.	80
Figura 3.21. Generación de reporte de diagnóstico.	80

Figura 3.22. Mensaje de notificación que el paciente no presenta diagnósticos.....	81
Figura 3.23. Acceso de usuario paciente.....	81
Figura 3.24. Visualización información personal paciente.	82
Figura 3.25. Sugerencia de cita médica.....	82
Figura 3.26. Carga de exámenes por parte del paciente.	83
Figura 3.27. Buzón de sugerencias y quejas.....	83
Figura 3.28. Encuesta a Usuario Médico.....	85
Figura 3.29. Encuesta a Usuario Paciente.	86
Figura 3.30. Integridad de información personal del paciente.....	90
Figura 3.31. Datos de rendimiento en Smartphone SONY.	92
Figura 3.32. Tablero final de Kanban.....	94

ÍNDICE DE TABLAS

Tabla 1.1. Ventas mundiales de <i>smartphones</i> a usuarios finales por sistema operativo. 11	11
Tabla 1.2. Comparación entre tipos de servicio en <i>Cloud Computing</i> 15	15
Tabla 1.3. Comparación tipos de <i>Cloud</i> 19	19
Tabla 1.4. Servicios de AWS. 20	20
Tabla 1.5. Servicios de GCP..... 21	21
Tabla.1.6. Servicios de CNT 22	22
Tabla 1.7. Comparación de Proveedores..... 23	23
Tabla 1.8. Comparación de bases de datos de tres proveedores de <i>Cloud Computing</i> ... 24	24
Tabla 2.1. Descripción de los requerimientos funcionales del médico. 32	32
Tabla 2.2. Descripción de los requerimientos funcionales del paciente. 33	33
Tabla 2.3. Descripción de los requerimientos no funcionales..... 33	33
Tabla 2.4. Características del Proveedor AWS..... 35	35
Tabla 2.5. Características del Servidor AWS. 35	35
Tabla 2.6. Descripción de los RF del médico, el subsistema asociado, y los casos de uso..... 38	38
Tabla 2.7. Descripción de los RF del paciente, el subsistema asociado y el caso de uso..... 39	39
Tabla 3.1. Descripción del RFM01 por comprobar funcionamiento..... 71	71
Tabla 3.2. Descripción de los RF asociados al SB02 por comprobar funcionamiento. ... 72	72
Tabla 3.3. Descripción de los RF asociados al SB03 por comprobar funcionamiento. ... 73	73
Tabla 3.4. Descripción de los RF asociados al SB04 por comprobar funcionamiento. ... 76	76
Tabla 3.5. Descripción de los RF asociados al SB05 por comprobar funcionamiento. 78	78
Tabla 3.6. SB06 por comprobar funcionamiento..... 81	81
Tabla 3.7. Lista de cumplimiento de los requerimientos funcionales. 84	84
Tabla 3.8. Respuesta a la Pregunta 1 del médico. 87	87
Tabla 3.9. Respuesta a la pregunta 1 del paciente..... 87	87
Tabla 3.10. Preguntas y respuestas a la encuesta usuario médico. 88	88
Tabla 3.11. Preguntas y respuestas a la encuesta usuario paciente. 88	88
Tabla 3.12 Requerimientos no funcionales. 89	89
Tabla 3.13 Respuesta a la Pregunta 2. 89	89
Tabla 3.14. Tiempo de realizar una acción. 91	91
Tabla 3.15. Rendimiento del sistema en varios modelos <i>smartphone</i> 92	92
Tabla 3.16 Lista de cumplimiento de los Requerimientos No Funcionales. 93	93

ÍNDICE DE CÓDIGOS

Código 2.1. Codificación de la tabla <i>Medico</i> en lenguaje <i>SQL</i>	57
Código 2.2. Ejemplo de consulta en <i>SQL</i>	57
Código 2.3. Código de Clase <i>Persona</i>	58
Código 2.4. Código de Clase <i>Medico</i>	59
Código 2.5. Código de conexión para la base de datos.....	59
Código 2.6. Código de la implementación de un Servicio Web.	60
Código 2.7. Código XML para la implementación gráfica.	63

RESUMEN

Desde el inicio de la medicina, médicos especialistas, centros de salud, hospitales, clínicas, etc., han tenido la necesidad de llevar la información de cada una de las personas que se acercan a ocupar un servicio de atención médica en un documento llamado “historial clínico”. En la actualidad a pesar de los grandes avances en tecnología hay especialistas en Traumatología en la clínica “Club de Leones Quito” que siguen guardando esta información en documentos escritos; un método que puede presentar problemas con el contenido, escritura ilegible, datos faltantes, formato pobre, dificultad en el acceso y la disponibilidad.

En base a estos antecedentes, el presente Proyecto de Titulación propone el diseño e implementación de una aplicación, para especialistas de traumatología del “Club de Leones Quito”, que permita crear, almacenar, editar y eliminar información necesaria de pacientes a través de un *smartphone* con sistema operativo Android. El almacenamiento de esta información será gestionado en un ambiente *Cloud Computing*, a través de la selección de un proveedor.

Así, el desarrollo del presente Proyecto de Titulación permitirá que especialistas en traumatología tengan un mejor manejo del historial clínico de sus pacientes, y así mismo agilizar el proceso ya que el paciente tendrá la opción de subir sus exámenes médicos al repositorio *cloud*. Todo esto con el propósito de cambiar el uso del método actual y hacer uso de la tecnología.

En el Capítulo 1, se describirá el objetivo general y el alcance que tendrá el presente Proyecto de Titulación, además; se describirá la situación actual de cómo es tratada la información de los pacientes en la clínica “Club de Leones Quito”. Finalmente, se menciona los fundamentos teóricos de *Cloud Computing*, servicios web, Android Studio y base de datos.

En el Capítulo 2, se describe la metodología para el desarrollo del presente Proyecto de Titulación, la cual será dividida en dos: diseño e implementación.

En el Capítulo 3, se verificará el correcto funcionamiento de Sistema de Gestión Móvil y se mostrará los resultados de las pruebas realizadas.

En el Capítulo 4, se describirán las respectivas conclusiones y recomendaciones.

PALABRAS CLAVE: Aplicación móvil, Computación en la Nube, *Smartphone*, Traumatología, Base de Datos.

ABSTRACT

From the beginning of medicine, medical specialists, health centers, hospitals, clinics, etc., they have had the need to bring the information of each of the people who come to occupy a health care service in a document called "medical history". Nowadays, despite the great advances in technology there are specialists in Traumatology in the clinic "Club de Leones Quito" who keep this information in written documents; a method that can present problems with content, illegible writing, missing data, poor format, difficulty in access and availability.

Based on this background, this Titling Project proposes the design and implementation of an application for trauma specialists of the "Club de Leones Quito", which allows creating, storing, editing and eliminating necessary patient information through a smartphone with Android operating system. The storage of this information will be managed in a Cloud Computing environment, through the selection of a provider.

Thus, the development of the present Titling Project will allow traumatology specialists to have a better management of the clinical history of their patients, and also expedite the process since the patient will have the option of uploading their medical exams to the cloud repository. All this with the purpose of changing the use of the current method and making use of technology.

In Chapter 1, the general objective and scope of this Titling Project will be described, in addition; The current situation of how patient information is treated in the "Club de Leones Quito" clinic will be described. Finally, the theoretical foundations of Cloud Computing, web services, Android Studio and database are mentioned.

In Chapter 2, the methodology for the development of this Titling Project is described, which will be divided into two: design and implementation.

In Chapter 3, the correct functioning of the Mobile Management System will be verified and the results of the tests carried out will be shown.

In Chapter 4, the respective conclusions and recommendations will be described.

KEY WORDS: Mobile application, Cloud Computing, Smartphone, Traumatology, Database.

1. INTRODUCCIÓN

El presente capítulo detalla en primera instancia los objetivos y el alcance que tendrá el Proyecto de Titulación; luego se describirá la situación actual de cómo es tratada la información de los pacientes por parte de los médicos del área de traumatología de la clínica “Club de Leones de Quito”. A continuación se presentará el análisis del mercado móvil de acuerdo a los sistemas operativos. Además se tratarán los fundamentos teóricos acerca de *Cloud Computing* y sus diferentes tipos de servicio y despliegue. De igual manera se describirán las herramientas y tecnologías para el desarrollo de este trabajo. Por último se estudiará la metodología Kanban la cual será utilizada en la realización de este Proyecto de Titulación.

1.1 Objetivos

El objetivo general de este Proyecto de Titulación es desarrollar un prototipo de un sistema de aplicación móvil de gestión médica para el área de traumatología en la clínica “Club de Leones Quito”

Los objetivos específicos de este Proyecto de Titulación son:

- Analizar los fundamentos teóricos necesarios para el desarrollo del sistema.
- Diseñar el prototipo de la aplicación.
- Implementar el prototipo con sus módulos.
- Analizar los resultados obtenidos en las pruebas realizadas.

1.2 Alcance

El presente Proyecto de Titulación propone el diseño e implementación de una aplicación para especialistas de traumatología del “Club de Leones Quito” que permita crear, almacenar, editar y eliminar información necesaria de pacientes a través de un *smartphone* con sistema operativo Android. El almacenamiento de esta información será gestionado en un ambiente *Cloud Computing* a través de la previa selección de un proveedor. Teniendo en cuenta que el médico dispondrá de la capacidad de: enviar, modificar o eliminar los exámenes médicos y el paciente solamente podrá enviar tales exámenes. De esta manera, y con el fin de agilizar el proceso de diagnóstico y mantener una historia clínica actualizada de los pacientes, los exámenes a más de observaciones

podrían ser fotografías y/o archivos. Sin embargo el análisis de dichos exámenes solamente podrá ser gestionado por el médico.

El especialista tendrá sus respectivas credenciales previo a un registro, esto como proceso de autenticación.

La Figura 1.1 muestra el esquema general del funcionamiento del prototipo, tanto para el desarrollo de la aplicación del médico, como la del paciente, donde las mismas estarán conectadas a una base de datos en la nube. A continuación se expone los procesos cronológicos que seguiría el prototipo a desarrollar:

- El paciente se acerca al traumatólogo y después de la respectiva revisión, el médico usará la aplicación la cual le permitirá crear una historia clínica en el caso de ser un nuevo paciente y en el caso de ser uno antiguo acceder a la información previamente guardada. Esta información contendrá datos personales, fechas de las visitas médicas y diagnósticos; los cuales permitirán conocer los avances en el estado de la salud del paciente.
- El médico podrá acceder a los historiales clínicos de sus pacientes desde cualquier parte y a cualquier hora, esto será muy útil en casos de emergencia. Se muestra que los datos y la parte del negocio serán ubicadas en la nube mientras que la interfaz gráfica estará en el dispositivo de cada paciente.
- La información para los requerimientos se obtendrá de encuestas al área de traumatología de la clínica “Club de Leones Quito” (7 médicos actualmente (2018))
- El administrador de la aplicación será el médico. Este podrá añadir, eliminar y modificar la información de cada uno de los pacientes que él dispondrá.

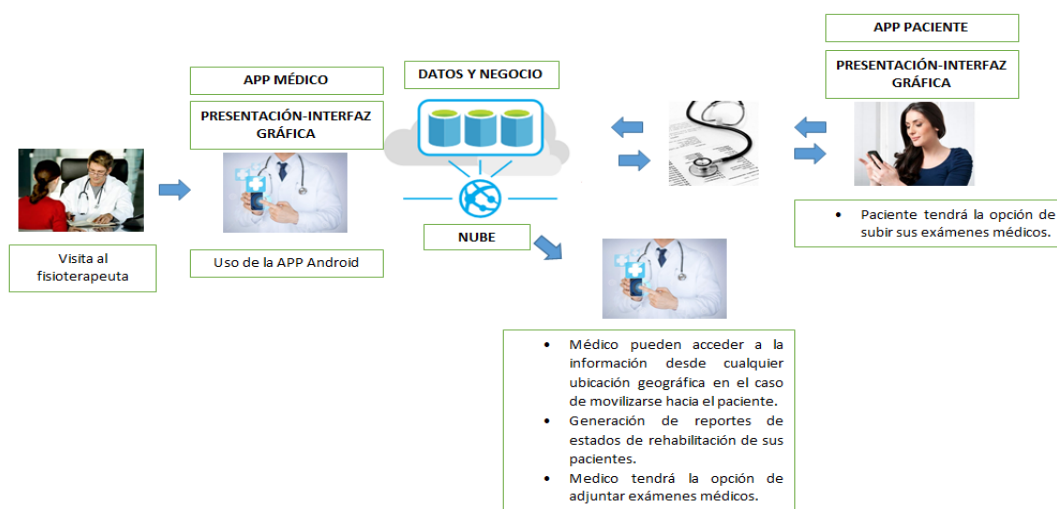


Figura 1.1. Esquema del Proyecto de Titulación.

Los requerimientos mínimos necesarios se describen con base en los dos actores principales:

- Administración del médico.
- Administración del paciente.

1.2.1 Administración del Médico

El sistema del médico tendrá los siguientes módulos mínimos:

- **Módulo de registro de médicos:** Se permitirá que los médicos puedan registrarse en el aplicativo.
- **Módulo de registro de pacientes:** El médico estará en capacidad de la administración de sus pacientes; ya sea añadir, modificar y eliminar información de los mismos.
- **Módulo de gestión de historias clínicas:** El médico estará en capacidad de la administración de los historiales clínicos de sus pacientes; ya sea añadir o modificar información de los mismos, pudiendo ser esta información también exámenes médicos.
- **Módulo de visualización de citas:** El médico podrá visualizar las próximas citas médicas. La información de las citas será previamente cargada. Teniendo en cuenta que este proceso lo realiza otro departamento del Hospital y no está contemplado en el plan de este Proyecto de Titulación aprobado.

1.2.2 Administración del Paciente

El sistema del paciente tendrá los siguientes módulos mínimos:

- **Módulo de acceso:** El paciente dispondrá de un *login* para su acceso hacia el sistema una vez que haya sido registrado por un médico.
- **Módulo de gestión de historias clínicas:** El paciente podrá enviar archivos de sus exámenes médicos al repositorio en la nube.
- **Módulo de información personal:** El paciente podrá visualizar la información personal. Cualquier cambio será realizado por el médico.
- **Módulos de sugerencias y quejas:** Se dispondrá de una opción donde los pacientes puedan comentar sus sugerencias o quejas para que el fisioterapeuta pueda realizar una retroalimentación de su servicio.

- **Módulo de visualización de citas:** El paciente podrá visualizar la sugerencia de la próxima cita médica.

1.3 Marco Teórico

Para llevar un registro de las atenciones médicas, dolencias, enfermedades y/o atenciones brindadas, es necesario para los profesionales en la salud registrar toda la información de sus pacientes en un documento llamado "historial clínico". El historial clínico al ser un documento importante debe tener ciertas características, a nivel de seguridad como por ejemplo autenticación, integridad, y almacenamiento seguro. Sin embargo, este documento generalmente es llevado en un archivo escrito o en un computador.

Por otro lado, en los últimos años tanto la salud como la tecnología han evolucionado y trabajado conjuntamente, esto ha hecho que la mayoría de médicos busquen en la tecnología un aliado. Además, puesto que el uso de *smartphones* ha tenido un gran crecimiento en los últimos años, y así también el gran despliegue de aplicaciones de todo tipo, haciendo de este dispositivo necesario en actividades cotidianas.

La inserción de aplicaciones en la salud ha revolucionado la relación médico-paciente y la relación de las personas con todo el sistema de salud, permitiendo así crear un modelo completamente nuevo en la medicina [1]. Por tanto, se ha hecho necesario el deseo de llevar la información a todo lado; poder acceder a ella en cualquier momento y lugar; además de que esta información debe estar almacenada en un lugar seguro.

Esto ha hecho que aparezca un concepto conocido ahora como Computación en la Nube (*Cloud Computing*) el cual permite visualizar y tener un concepto de que se tiene una gran nube donde todo se encuentra interconectado y se puede acceder a todos los servicios requeridos [2].

En consecuencia, el presente Proyecto de Titulación permitirá desarrollar un sistema de gestión médica móvil, que posibilitará al médico, en este caso en traumatología, llevar la información de los pacientes en una aplicación instalada en su *smartphone*. Además, el sistema dispondrá de la aplicación para el paciente, que le permitirá enviar exámenes médicos, visualizar sugerencias de visitas y observar su información personal, permitiendo tener una comunicación entre el médico y el paciente

Este prototipo tiene como objetivo que el médico pueda acceder a los tratamientos de rehabilitación de sus pacientes en una forma eficiente. La gestión móvil permitirá al especialista mantener la información de una manera ubicua.

1.3.1 Estado del Arte

a. Situación Actual del Manejo de Historiales Clínicos del Área de Traumatología de la Clínica “Club de Leones Quito”

La clínica “Club de Leones Quito” [3], es una entidad sin fines de lucro destinada a brindar ayuda al público en general; esta ayuda es otorgada a través de servicios de salud, educación, voluntariado, brigadas médicas y obras comunitarias. Las políticas de eficacia que maneja esta institución son calidad y satisfacción al cliente.

Al ser una institución médica con múltiples especialidades dispone de muchos médicos y pacientes, por tal motivo es importante que la información de los especialistas e historiales clínicos sea tratada de forma segura y ordenada. El historial clínico lleva la información de rehabilitación o tratamientos de cada uno de los pacientes que son atendidos. El historial clínico actualmente se encuentra en carpetas y documentos escritos, los cuales pueden ser propensos a pérdidas o daños. Los exámenes médicos son llevados por los pacientes en muchos de los casos impresos y puede darse el problema de que se pierdan los registros pasados.

Dentro de los servicios médicos que ofrece la institución se tiene los siguientes departamentos: cirugía, clínica, diagnóstico, tratamiento, chequeo ejecutivo y emergencias. El área de traumatología pertenece al Departamento de Cirugía de la institución, la cual está compuesta de 7 especialistas.

b. Otros Proyectos en Traumatología

La salud y la tecnología en los últimos años han tenido una estrecha relación, es por eso que tanto médicos como ingenieros han visto la necesidad de unir esfuerzos. Esto con el fin de desarrollar aplicaciones que permitan una mejor interacción del médico y el paciente.

A continuación se detalla algunos proyectos relevantes y el uso de aplicaciones en el área de traumatología:

b.1. Desarrollo de una aplicación móvil para manejo de sucesos críticos durante la cirugía ortopédica y traumatológica [4].

La revista colombiana de Ortopedia y Traumatología [4], en una de sus ediciones realizó una publicación en la cual detalla la iniciativa de desarrollar un software en forma de aplicación para teléfonos inteligentes o *tablets*, que permitiera afrontar los sucesos que puedan darse en una cirugía ortopédica; esto ya que se pueden dar múltiples sucesos

críticos intraoperatorios, pudiendo marcar una diferencia sustancial entre la vida y la muerte.

En el artículo se menciona que se puede dar 145 sucesos críticos por cada 10.000 procedimientos y los errores médicos son la principal causa de lesión inevitable de muerte en pacientes durante la cirugía.

El desarrollo de esta aplicación para plataformas Android e iOS, que utiliza POP (*Prototyping on Paper*)¹, es una aplicación que contiene un diagrama de flujo el cual permite la identificación temprana de un suceso crítico durante la cirugía y, una vez detectado provee los pasos necesarios para un adecuado afrontamiento.

Las pruebas fueron realizadas en un ambiente de simulación efectuada en diferentes salas de cirugía donde se probó la utilidad de la aplicación y el diagrama. La revista indica además que hasta la fecha no existe una herramienta similar, lo que demuestra que es necesario incursionar en el desarrollo de este tipo de aplicaciones.

El uso de la aplicación mejoró los tiempos de respuesta y los médicos refirieron comodidad con el uso de la aplicación. En la Figura 1.2 se muestra un esquema general de esta aplicación.



Figura 1.2. Esquema de la Aplicación de Procedimientos Quirúrgicos.

La Figura 1.3 muestra el diagrama de flujo el cual permite mostrar la secuencia de los pasos de cómo debe ir actuando el especialista en el caso de presentarse algún suceso crítico durante la cirugía y, una vez detectado, provee de los pasos para su adecuado afrontamiento basado en situaciones similares anteriores. El diagrama de flujo permite que se haga un mejor uso de la aplicación.

¹ POP convierte los wireframes dibujados a mano en prototipos interactivos.

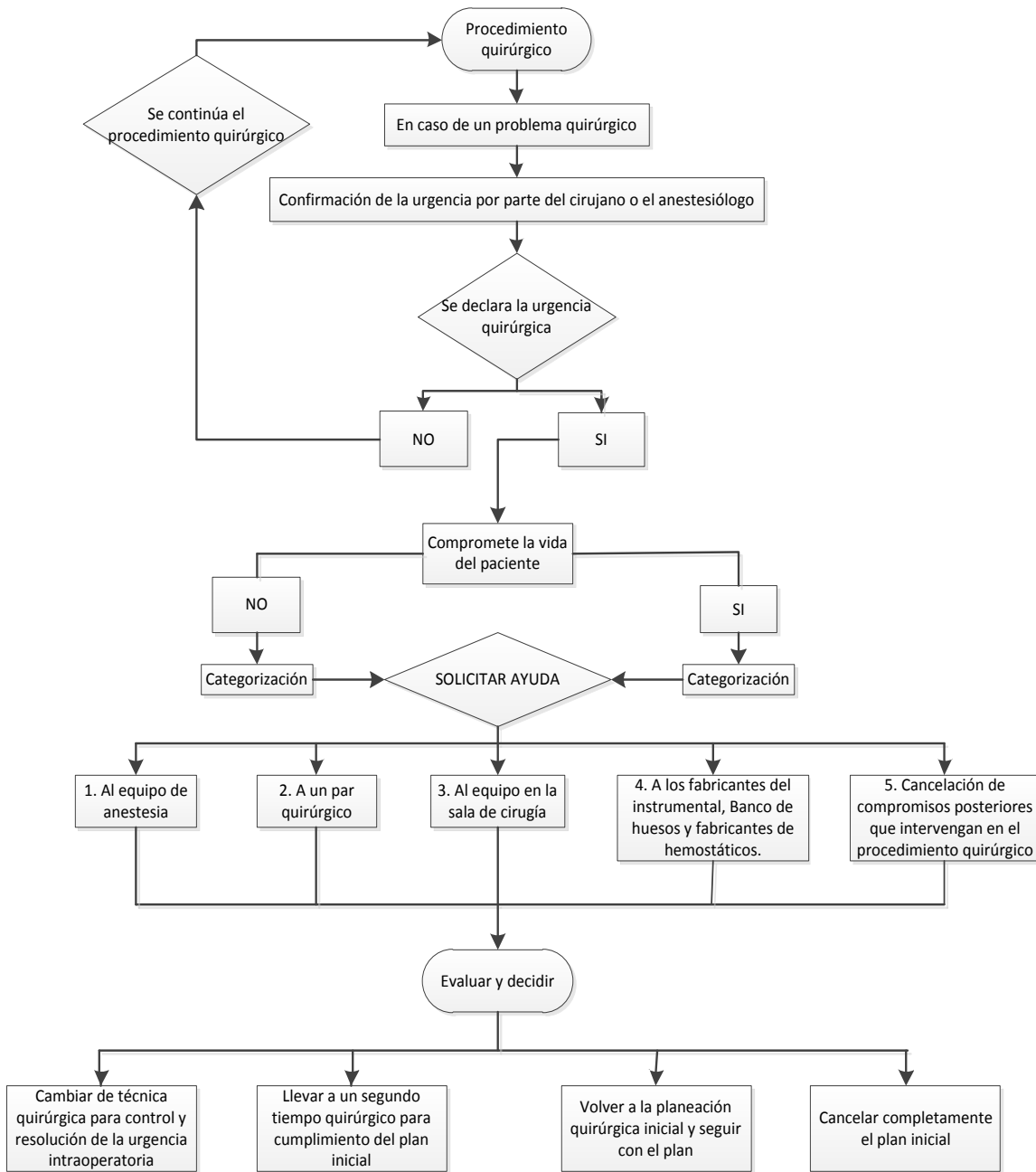


Figura 1.3. Diagrama de flujo del funcionamiento de la aplicación de Procedimientos Quirúrgicos[4].

Este artículo ha permitido que este tipo de aplicaciones sean de gran ayuda probando su utilidad en diversas áreas del conocimiento. La revista cita: *“iniciativas como la propuesta tienen el potencial de mejorar la seguridad del paciente y son un ejemplo de innovación para el mundo”*.

b.2. ICUC: La app uruguaya para medicina interactiva [5].

Esta aplicación tiene como objetivo que tanto estudiantes como profesionales en el área de traumatología y la ortopedia puedan buscar de primera mano diagnósticos, cirugías y

tratamientos postoperatorios; dejando atrás el mito de que todo lo que pasa en medicina debe estar escrito en alguna revista especializada y, así esta información pueda ser accedida por los médicos a través de una aplicación.

“Lo que yo veo tú lo ves” (*I see you see*) es el nombre que lleva esta aplicación desarrollada por un equipo de Uruguay. Es una aplicación para tablets donde se dispone de más de 1000 casos de fracturas con información de diagnóstico hasta el postoperatorio que se encuentran almacenados en una base de datos, y permiten llevar de mejor manera la evolución de estado de salud del paciente. La aplicación, además dispone de más de 10000 placas, fotografías, videos y animaciones en 3D que permitirá recrear las formas en la que se realizó la operación.

La Figura 1.4 muestra que luego de un previo registro en la aplicación, el traumatólogo puede acceder a toda la información y cientos de casos de operaciones que están incluidas en esta herramienta. Para comenzar puede elegir entre miembros superiores e inferiores. Y además la aplicación cuenta la posibilidad de activar un tutorial.



Figura 1.4. Esquema de aplicación ICUC.

Esta aplicación señala que solo a través de la innovación es posible mejorar la calidad de la medicina.

b.3. Fiabilidad de un goniómetro de un teléfono inteligente (*smartphone*) para la goniometría de la articulación en la rodilla [6].

Un goniómetro universal² es utilizado como método general para evaluar el rango de movimiento en articulaciones. Dado que el goniómetro es uno de los sensores que utilizan la mayoría de teléfonos en la actualidad, ha hecho que este sensor sea también

² Instrumento que sirve para medir ángulos, es un aparato en forma de semicírculo graduado en 180 y 360 grados.

utilizado como aplicación para los médicos traumatólogos en el proceso de recuperación de sus pacientes.

Este proyecto está basado en comparar un dispositivo goniómetro con otro instalado en un teléfono. Esta aplicación *mHealth*³ puede ser un método alternativo/adicional de goniometría a métodos tradicionales basados en fotografías. El método utilizado por los especialistas trata que mediante fotografías se evalúe el proceso de rehabilitación de un paciente. En primer lugar la imagen debe descargarse de la cámara a un ordenador y a continuación, debe ser editada utilizando un programa informático específico. Dado que éste puede ser un proceso largo y complicado, por lo tanto representa una desventaja, ésta puede ser eliminada gracias a la nueva generación de *smartphones*.

La Figura 1.5 muestra el uso de la aplicación instalada en un teléfono, que con ayuda de la propia cámara y del goniómetro hace mucho más fácil el proceso de diagnóstico y tratamiento de un paciente dejando atrás procesos obsoletos.

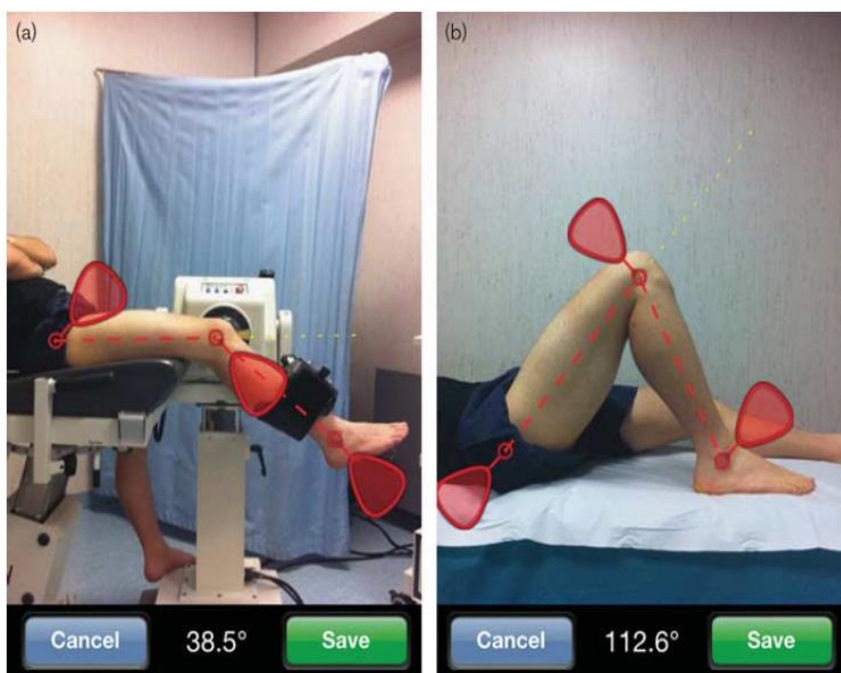


Figura 1.5. Uso de la aplicación en un tratamiento de rehabilitación [6].

La aplicación de procedimientos quirúrgicos en casos de emergencia, ICUC, y la aplicación de uso del goniómetro en dispositivos móviles son ejemplos que muestran la existencia de proyectos en salud y específicamente para el área de traumatología. Se

³ *mHealth* es un término acuñado por la Organización Mundial de la Salud que lo define como la práctica de la medicina apoyada en dispositivos móviles.

indica que el uso de las aplicaciones para esta área resulta ser muy útil y facilita al médico en sus actividades aprovechando el uso de la tecnología.

1.3.2 Smartphones y Aplicaciones Móviles

El uso de dispositivos móviles ha crecido a gran escala en los últimos años, esto se ha dado ya que una de las características en este momento es la movilidad, haciendo de estos dispositivos, indispensables en la vida cotidiana del ser humano. Los *smartphones* se han convertido en parte importante en las actividades diarias.

Un dispositivo móvil se puede definir como un dispositivo de pequeño tamaño con capacidad de procesamiento, conexión a la red y con memoria limitada que permite una movilidad física. Este concepto permite que los usuarios puedan acceder a la información, datos u otros objetos lógicos desde cualquier lugar y en cualquier red mientras están en movimiento [7].

La potencia de cómputo de un *smartphone*, actualmente es comparable a la de un ordenador de escritorio o portátil, ya que son capaces de ejecutar un sistema operativo móvil generando posibilidades hasta años atrás no pensadas [8].

Además, la creciente demanda de aplicaciones específicas para *smartphones* ha hecho que los desarrolladores tengan nuevos desafíos, ya que estas deben tener sus propias características y restricciones únicas, lo que ha hecho que se diferencie del desarrollo de software tradicional. Por estas razones es que en las tiendas de *Google Play* [9] o *App Store* [10] se encuentra una gran variedad de aplicaciones de todo tipo haciendo que este mercado este sujeto a cambios radicales en los últimos años y de la misma manera se creen diferentes alternativas de desarrollo de una misma aplicación para un dispositivo móvil[11].

Por otro lado, en los últimos años los avances tecnológicos han permitido diversificar los dispositivos para acceder a internet con el objetivo de permitir llevar la información a todos los usuarios en cualquier momento y en cualquier lugar. Además, se ha presenciado la evolución de las tecnologías referentes a servicios de telefonía, las telecomunicaciones, almacenamiento, hardware, software, etc. [7].

a. Análisis del Mercado Móvil

El desarrollo de aplicaciones en los últimos años ha crecido de forma exponencial, y esto ha llevado que los sistemas operativos más populares tengan de igual manera un crecimiento fuerte, teniendo como líderes a la plataforma de Google con Android, seguida de Apple con iOS. Según estadísticas hasta febrero del 2017 [12] indican que estas dos

plataformas representan el 99.6% de la participación del mercado total. La Tabla 1.1. muestra la cantidad de ventas en los dos últimos años, y demuestra que el sistema operativo Android es el de mayor uso y comercialización, por tal razón este Proyecto de Titulación será desarrollado para Android con el fin de abarcar con la mayor cantidad de usuarios, bajo un sistema de aplicaciones nativas, que son aquellas que permiten aprovechar la ejecución en una plataforma específica. Para esto se debe tomar en cuenta el tipo de dispositivo que se va a utilizar, el sistema operativo y su versión. Se debe tomar en cuenta que una de las ventajas que se tiene es el ocupar las funciones del dispositivo.

Tabla 1.1. Ventas mundiales de *smartphones* a usuarios finales por sistema operativo[12].

Sistema Operativo	Número de unidades en 2015	Porcentaje de participación %	Número de unidades en 2016	Porcentaje de participación %
Android	325.394	80,7	352.669	81,7
IOs	71.525	17,7	77.038	17,9
Windows	4.395	1,1	1.092	0,3
BlackBerry	906	0,2	207	0,0
Otros sistemas operativos	887	0,2	530	0,1
Total	403.109	100,0	431.539	100,0

Según la OMS (Organización Mundial de la Salud)[13], el desarrollo de aplicaciones móviles está destinado directa o indirectamente a mantener o mejorar la calidad de vida y el bienestar de las personas. Se estima que en las tiendas de aplicaciones existen alrededor de 97.000 aplicaciones de salud, y esta cifra podría tener un crecimiento del

23% anual en los próximos 5 años [14]. La Figura 1.6. muestra el tipo de aplicación y el porcentaje de utilización que tienen los usuarios hacia las aplicaciones médicas [15].

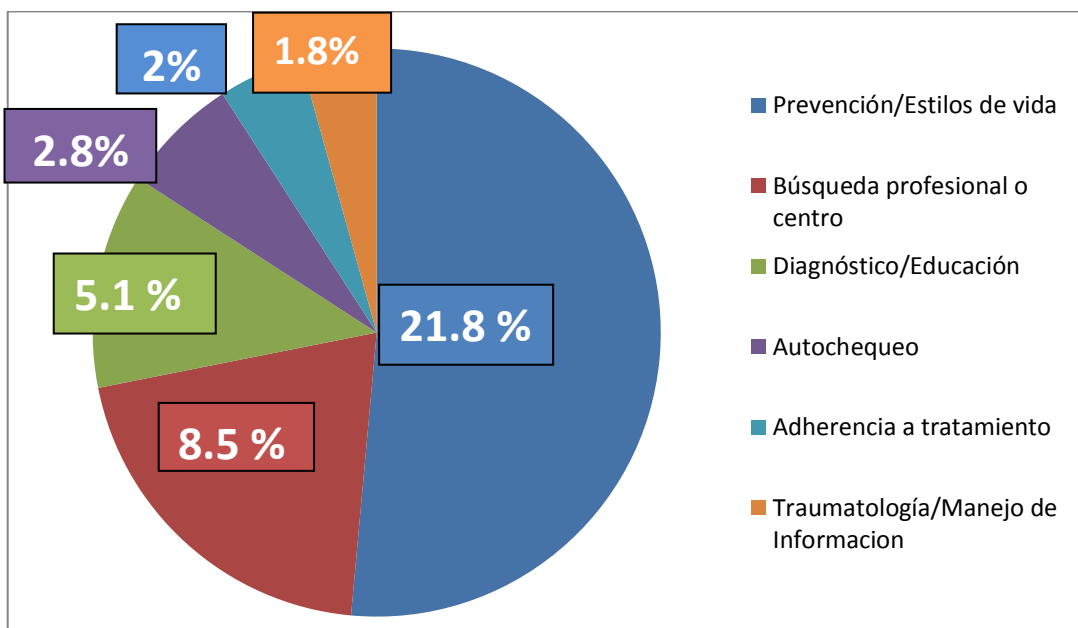


Figura 1.6. Tipo de aplicaciones médicas y porcentaje de utilización[15].

1.3.3 Cloud Computing

La Computación en la Nube (*Cloud Computing*), es un modelo que permite el acceso a aplicaciones y servicios *on-line* a través de cualquier dispositivo que tenga acceso a Internet sin necesidad de disponer, ya sea licencias de software o grandes recursos en hardware [2].

La Computación en la Nube tiene el potencial de transformar una gran parte de la industria de la información y de las comunicaciones, haciendo que el software sea aún más atractivo como servicio y así mismo el hardware sea aprovechado de una mejor manera. Puesto que nuevas ideas hacen uso de los servicios de Internet, ya no se requieren de grandes desembolsos de capital en hardware para implementar su servicio o el gasto humano para operarlo. La Computación en la Nube es un servicio ubicuo, seguro y bajo demanda [16].

a. Clasificación por Tipo de Servicio

El primer tipo de servicio es el de **Infraestructura como Servicio (IAAS)** [17] el cual consiste en que un cliente haga uso de la **Infraestructura como Servicio** ya sea ésta: capacidad de computación, espacio de discos y redes; en sí la parte de hardware.

Los clientes toman esta opción para evitar grandes gastos en recursos físicos, ya que todos estos estarán para el servicio de los clientes de acuerdo a los requerimientos del

proyecto y necesidades. Esto hace que el costo se calcule en base a los recursos consumidos por el cliente, basándose así en una de las características de pago por uso.

La Figura 1.7 muestra ejemplos de lo que brinda un tipo de servicio IAAS. Como se puede observar IAAS ofrece al usuario todo lo referente a recursos físicos haciendo que el cliente no deba realizar grandes inversiones en la compra e instalación de sus equipos. El proveedor de servicios IAAS se encargará de todos estos detalles, además de que el pago se realizará de acuerdo a lo que esté siendo utilizado por parte del cliente. El tipo de servicio IAAS maneja la parte de virtualización, servidores, almacenamiento y redes [17].

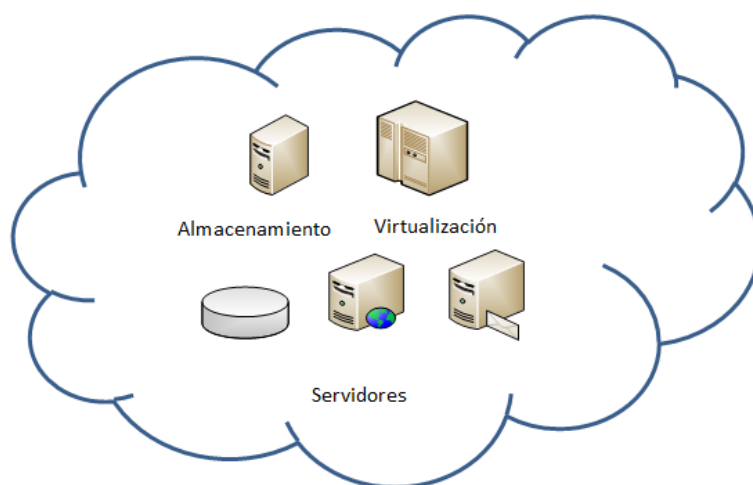


Figura 1.7. Ejemplos de tipo de servicio IAAS.

El segundo tipo de servicio es el de **Plataforma como Servicio (PAAS)** el cual consiste en entregar una Plataforma de Desarrollo, con el fin de que los programadores en la web utilicen ésta como un servicio. Principalmente se entrega herramientas tipo *middleware*⁴, por ejemplo: Google AppEngine, que permite al desarrollador diseñar, implementar y probar sus proyectos todo, en un ambiente de *Cloud Computing*. El cliente final no administra la infraestructura de la nube; incluida la red, los servidores, los sistemas operativos o el almacenamiento, pero si tiene control sobre las aplicaciones desarrolladas e implementadas y si es el caso de las configuraciones para el entorno de alojamiento de la aplicación [16].

En la Figura 1.8 se muestra los ejemplos de un servicio tipo PAAS, entre los principales proveedores se pueden mencionar a Google AppEngine [18], Windows Azure [19], y Amazon Web Services (AWS) [20].

⁴ Lógica de intercambio de información entre aplicaciones, es un software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones.



Figura 1.8. Ejemplos de tipo de servicio PAAS.

El tercer tipo de servicio es el **Software como Servicio (SAAS)** el cual consiste en que el ambiente de *Cloud Computing* entregue a los usuarios finales aplicaciones como servicios. Es un simple concepto para el usuario final, ya que éste ingresa a una aplicación a través del navegador web sin saber dónde se encuentra o cómo funciona; solamente sabe que es rápida y segura. La Figura 1.9 muestra algunos ejemplos de SAAS como: correo electrónico, almacenamiento de archivos, trabajos *online*, Office 365.



Figura 1.9. Ejemplo de tipo de servicio SAAS.

El usuario en este tipo de servicio es capaz de acceder a todas las aplicaciones que ofrece un proveedor, accediendo a las mismas desde cualquier dispositivo con acceso a Internet. El usuario no administra ni controla la infraestructura, almacenamiento, servidores, y sistemas operativos [16].

La Tabla 1.2 muestra una comparativa entre los diferentes tipos de servicios que se tiene en un ambiente *Cloud Computing*.

Tabla 1.2. Comparación entre tipos de servicio en *Cloud Computing*.

IAAS	PAAS	SAAS
Servicios de computación optimizados a bajo costo. El costo dependerá del uso de la infraestructura.	Optimización en costos con referente a licencias de programas a utilizarse en el desarrollo.	Ahorro de inversión en infraestructura. Solo se necesita una conexión a Internet.
Administración compartida entre el usuario y el proveedor.	Administración de las aplicaciones y datos por parte del usuario.	El proveedor se encarga del mantenimiento y funcionamiento del Software.
La seguridad es completamente realizada por el proveedor	Gestión de seguridad y administración por parte del usuario.	La seguridad es controlada por el administrador del servicio.
Brinda los recursos como servidores, discos duros, entre otros, según la necesidad del usuario.	Tipo de servicio multiusuario, permite tener varios usuarios, manteniendo la escalabilidad y seguridad del sistema.	Es totalmente escalable y dimensionable. Cliente solo accede a su aplicación favorita a través de la red.

b. Clasificación por Tipo de Despliegue

La clasificación del *Cloud Computing* por tipo de despliegue se tiene en primer lugar el **Cloud Público** o **Nube Pública** que es una expresión que hace referencia a la infraestructura que puede hacer uso el público en general, como grandes y medianas empresas, a través de Internet; y es ofrecida por un proveedor que comercializa servicios bajo demanda. Esta forma de clasificación se caracteriza por los servicios virtualizados, estos pueden ser bases de datos, sistemas operativos, plataformas de desarrollo por parte de los proveedores para múltiples clientes, accediendo a estos solamente con una conexión a internet o a través de redes privadas virtuales (VPN) [16]. Entre los principales proveedores de **Cloud Público** se puede nombrar a Microsoft Azure [19], Oracle [21], AWS [20], Google [18], IBM [22] [23].

La Figura 1.10 muestra un esquema donde los usuarios quieren acceder a utilizar los servicios que se presta en un ambiente de *Cloud* a través de un proveedor. Estos proveedores pueden facilitar servicios de tipo IAAS, PAAS, SAAS. Generalmente este

modelo de despliegue se relaciona con un contexto comercial, donde los usuarios son considerados clientes y pagan por el tiempo que estén usando los servicios.

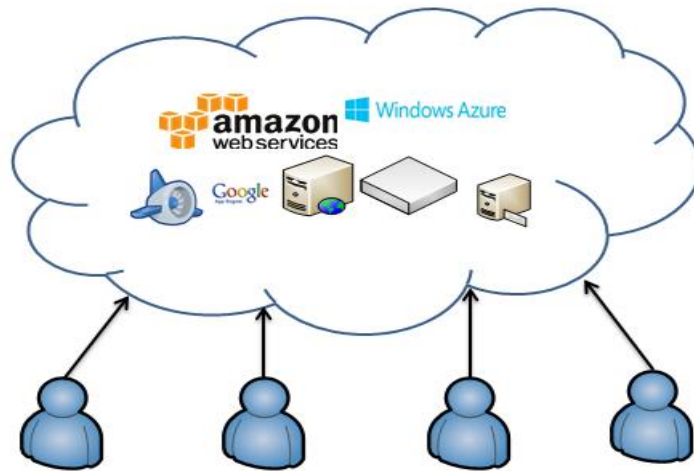


Figura 1.10. Cloud Público.

A continuación, se tiene el **Cloud Privado (Interno)** o **Nube Privada** que se define como la infraestructura que ofrece servicios informáticos a través de Internet solo a algunos usuarios y no al público en general.

Esta forma de implementación se caracteriza en que la organización o empresa tiene su propio ambiente de *Cloud Computing* dentro de su intranet, haciendo que los entornos virtualizados sean implementados, usados y controlados por la misma organización evitando así que agentes externos accedan a la información, generando así una sensación de mayor seguridad. Sin embargo, la desventaja principal es que las organizaciones están expuestas a grandes inversiones para la implementación de un **Cloud Privado**.

La Figura 1.11 muestra que el tipo de despliegue es realizado sobre la infraestructura de hardware propia de la organización, haciendo que los servicios que presta sean de similares características que la de un entorno **Cloud Público** con la diferencia de que el acceso está limitado a los usuarios de la red privada. Además, la nube privada puede tener un nivel más alto en la parte de seguridad y privacidad con *firewalls*⁵ propios de la compañía, impidiendo que agentes externos accedan y así garantizar integridad en los datos de la empresa.

⁵ Programa informático que controla el acceso de una computadora a la red y de elementos de la red a la computadora, por motivos de seguridad.



Figura 1.11. Cloud Privado.

También, otra forma de tipo de despliegue es una implementación de *Cloud* que está interconectada entre *Cloud* públicos y privados, haciendo uso de infraestructura estandarizada o propietaria. Al igual que un *Cloud* privado o público, un **Cloud Híbrido** permite la portabilidad de datos y aplicaciones. Es un servicio de *Cloud* integrado que permite realizar diferentes funciones dentro de una misma organización.

La Figura 1.12 es un ejemplo de cómo se compone un **Cloud Híbrido**, los cuales están compartidas en tecnología estandarizada o patentada que permite la portabilidad de datos y aplicaciones.

En el **Cloud Híbrido** la organización proporciona y gestiona algunos recursos internos y algunos externos. El enfoque principal de un sistema híbrido es permitir que la organización aproveche la escalabilidad y la rentabilidad que ofrece un entorno *Cloud* público y de la misma manera aprovechar equipos propios de la empresa [24].

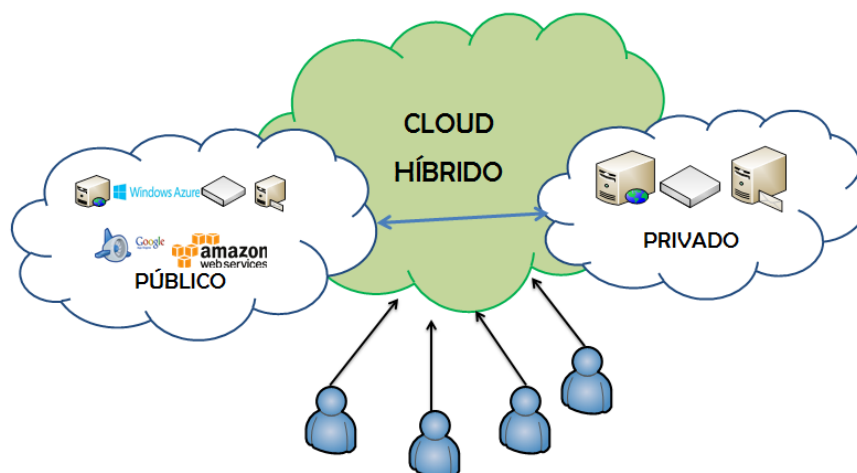


Figura 1.12. Cloud Híbrido.

Finalmente, se tiene un **Cloud Comunitario** o **Nube Comunitaria** que es aquella en donde la infraestructura tecnológica es implementada por diferentes organizaciones o empresas cuyos objetivos, funciones y servicios son comunes; además pueden establecerse en las propias instalaciones del grupo o fuera de ellas.

Algunos ejemplos de este tipo de *Cloud* se pueden nombrar en la salud donde las instituciones médicas tienen acceso a aplicaciones e información de carácter sanitario. También, se puede observar en las instituciones gubernamentales donde comparten información de la administración pública o en instituciones educativas donde administrativos y estudiantes pueden acceder a los diferentes recursos. Este tipo de nube puede ser administrada por las propias organizaciones o empresas, o a su vez por terceros.

Un **Cloud Comunitario** está constituido en su mayor parte de un **Cloud Híbrido** con la diferencia de que todos los recursos computacionales son exclusivos de dos o más organizaciones por lo que el número de usuarios es menor. Dentro de las consideraciones comunes que manejan estas organizaciones se ofrece un mayor nivel de privacidad, seguridad y regulación [24].

La Figura 1.13 muestra a un **Cloud Comunitario** conformado por cuatro organizaciones, cada una de ellas tiene acceso a los recursos dispuestos para su uso común.

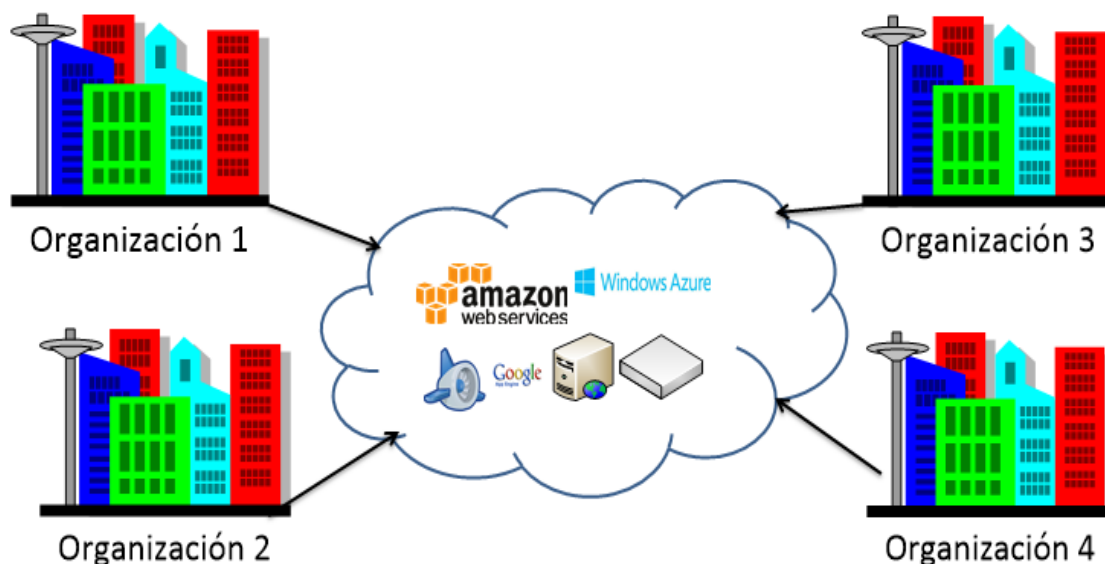


Figura 1.13. Cloud Comunitario.

La Tabla 1.3 muestra una comparativa entre los diferentes tipos de *Cloud* que existen en el mercado.

Tabla 1.3. Comparación tipos de *Cloud*.

Cloud Pública	Cloud Privado	Cloud Híbrido	Cloud Comunitario
Facilidad de escalamiento, y costes más bajos	Dificultad en el escalamiento y costes más altos.	Características tanto de un <i>Cloud Pública</i> y un <i>Cloud Privado</i>	Facilidad de escalamiento y costes más bajos.
La seguridad y la calidad en manos del proveedor de <i>Cloud</i> .	La seguridad de los archivos depende de red y el administrador de la misma.	La seguridad y la calidad están divididas entre el administrador de la red y el proveedor.	La gestión es hecha por una de las organizaciones o fuera de ellos.
Tiene un mayor nivel de autogestión.	Riesgo de obsolescencia.	Posibilidad y facilidad de mayor crecimiento en el <i>Cloud Pública</i> .	La privacidad y la seguridad es a cargo del proveedor.
Proyectos con posibilidad de crecimiento.	Proyectos a medida	Adecuado en escenarios con proyectos con futuro crecimiento.	Permite el crecimiento de cada organización.
Abierta a todo el que realice un pago o una suscripción.	Exclusivamente para una organización.	Organización en crecimiento que desea tener información compartida.	Compartido por varias organizaciones.

c. Proveedores Públicos

Los proveedores de *Cloud Computing* forman parte importante de todo el proceso de implementación de servicios en la nube. A continuación se presenta dos proveedores de Servicios internacionales y uno nacional.

En primer lugar se tiene a Amazon Web Services (AWS) [20], que es un IASS que ofrece servicios de infraestructura como potencia de cómputo, almacenamiento de base de datos, entrega de contenido; donde el principal objetivo es que los recursos puedan escalar y crecer.

La nube de AWS funciona en 49 zonas de disponibilidad dentro de 18 regiones geográficas del mundo, con planes anunciados para crear 12 zonas más en Baréin, China, Francia, Hong Kong, Suecia y una segunda región en los EE. UU [20].

La Tabla 1.4. presenta una gran variedad de productos y servicios que ofrece AWS. A continuación, se mencionan las líneas más principales:

Tabla 1.4. Servicios de AWS[20].

Computación y almacenamiento	<ul style="list-style-type: none"> • Almacenamiento escalable en la nube. • Servidores virtuales en la nube. • Almacenamiento y recuperación de imágenes Docker. • Ejecute y administre contenedores de Docker⁶.
Redes y entrega de contenido	<ul style="list-style-type: none"> • Red de entrega de contenido global. • Sistema de nombres de dominio escalable. • Equilibrio de carga de gran escala.
Herramientas para desarrolladores	<ul style="list-style-type: none"> • Desarrollo de aplicaciones de AWS. • Almacenamiento de código en repositorios privados. • Compilación y prueba de código. • Automatización de la implementación de código.
Seguridad, identidad y conformidad	<ul style="list-style-type: none"> • Administración en el acceso de usuarios. • Analizar la seguridad de las aplicaciones. • Aprovisionamiento, administración e implementación de certificados SSL/TLS.
Análisis e inteligencia artificial	<ul style="list-style-type: none"> • Creación de bots conversacionales de voz y texto. • Análisis facial y de imágenes. • Aprendizaje automático para desarrolladores.
Servicios móviles y servicios de aplicaciones	<ul style="list-style-type: none"> • Kit de desarrollo de software móvil. • Notificaciones push para aplicaciones móviles. • Crear, probar, y monitorear aplicaciones. • Crear, implementar y gestionar APIs.
Internet de las cosas (IoT)	<ul style="list-style-type: none"> • Conexión de dispositivos a la nube. • Computación local, mensajería y sincronización para dispositivos. • Dash Button programable en la nube.
Desarrollo de videojuegos	<ul style="list-style-type: none"> • Hospedaje de servidores de videojuegos dedicados simples, rápidos y rentables. • Un motor de juegos gratuito multiplataforma en 3D con código fuente completo.

⁶ Docker es un proyecto de código abierto que permite crear “contenedores” (máquinas virtuales ligeras).

En segundo lugar se tiene a Google Cloud Platform (GCP), que es propiedad de Google quien ha decidido reunir todas las aplicaciones que ofrecía por separado en una plataforma denominada Google Cloud Platform. GCP ofrece servicios para poder crear soluciones a través de la tecnología que tienen en la nube, como por ejemplo se puede destacar la rapidez y la escalabilidad que posee Google en las aplicaciones de su buscador.

GCP ofrece servicios de tipo SaaS, PaaS y IaaS, lo que permite a los usuarios tener una gran gama de herramientas, permitiendo a los desarrolladores no tener que preocuparse ya sea en recursos de infraestructura o software, facilitando así el trabajo. GCP dispone de ciertos servicios que se listan en la Tabla 1.5.

Tabla 1.5. Servicios de GCP [18].

Recursos Informáticos	<ul style="list-style-type: none"> • Máquinas virtuales. • Plataforma como servicio para aplicaciones. • Despliegue de contenedores • Entorno sin servidor para crear y conectar servicios en la nube.
Almacenamiento y bases de datos	<ul style="list-style-type: none"> • Almacenamiento de objetos • Servicios de bases de datos MySQL y PostgreSQL • Servicio de bases de datos relacional. • Almacenamiento en bloque de instancias de máquina virtual.
Redes	<ul style="list-style-type: none"> • Redes VPC para recursos de Google <i>Cloud Platform</i>. • Balanceo de carga escalable de alto rendimiento. • Distribución de contenido en la red global de Google.
Big Data	<ul style="list-style-type: none"> • Almacenamiento de datos empresariales rápido. • Servicio de procesamiento de datos totalmente administrado. • Servicio Spark y Hadoop administrado, rápido y fácil de usar.
Internet de las Cosas (IoT)	<ul style="list-style-type: none"> • Plataforma que se encarga de obtener información de todos los dispositivos conectados a la nube.

Por otro lado, la Corporación Nacional de Telecomunicaciones (CNT), en Ecuador ofrece el servicio de *Cloud Computing*. El servicio de *Cloud* de CNT, es escalable, es decir que el cliente final puede pedir los recursos necesarios de acuerdo al crecimiento que tenga.

La nube que ofrece CNT actualmente es la única que cumple con la norma internacional ISO 270001 [25], que es un estándar en seguridad de la información [26].

La Tabla.1.6 muestra los servicios que brinda CNT para el manejo de información en la nube.

Tabla.1.6. Servicios de CNT [26].

Centro de datos virtuales	<ul style="list-style-type: none"> • Servicio de alta confiabilidad y disponibilidad. • Implementación de recursos de cómputo de forma rápida a través de la nube. • La posibilidad de agregar servicios adicionales como seguridad, soporte, capacitación, monitoreo y software.
Comunicaciones Unificadas	<ul style="list-style-type: none"> • Servicios de voz, video, mensajería instantánea a través de la nube CNT. • Cifrado de la información para garantizar seguridad en la comunicación.
Correo Electrónico CNT	<ul style="list-style-type: none"> • Servicio de correo con una capacidad de hasta 5 GB de almacenamiento por buzón.
Centro de datos físicos	<ul style="list-style-type: none"> • Data Centers en Quito y Guayaquil. • Sistemas redundantes • Estricto Sistema de seguridad.
<i>Web Hosting</i>	<ul style="list-style-type: none"> • Almacenamiento de datos, imágenes, sonidos y todo el contenido en la nube empresarial.

A continuación, se presenta una comparación entre los proveedores mencionados. Los proveedores de *Cloud Computing* poseen grandes tecnologías, dando así múltiples beneficios al usuario y dejando una difícil decisión al momento de escoger una opción. La

Tabla 1.7 muestra una comparación entre los 3 proveedores mostrando las principales características entre ellos.

Tabla 1.7. Comparación de Proveedores.

	Amazon Web Services	Google Cloud Platform	CNT
Escalabilidad automática	<ul style="list-style-type: none"> • Si 	<ul style="list-style-type: none"> • Si 	<ul style="list-style-type: none"> • SI
Soporta sistema operativo Windows	<ul style="list-style-type: none"> • Windows Server 2003 R2 • Windows Server 2008 • Windows Server 2012 	<ul style="list-style-type: none"> • Windows Server 2003 • Windows Server 2008 	<ul style="list-style-type: none"> • Windows Server 2003
Soporta sistema operativo Linux	<ul style="list-style-type: none"> • SI 	<ul style="list-style-type: none"> • SI 	SI
Soporte para lenguajes	<ul style="list-style-type: none"> • C++ • C# • Java • Perl • Python 	<ul style="list-style-type: none"> • Python • Java 	<ul style="list-style-type: none"> • Java
Soporte para almacenamiento de datos	<ul style="list-style-type: none"> • SI 	<ul style="list-style-type: none"> • SI 	<ul style="list-style-type: none"> • SI

d. Base de Datos en un Entorno *Cloud* [27]

Como se ha estudiado en el **Apartado a** del presente Capítulo existen diferentes arquitecturas de servicio; IAAS, PAAS y SAAS. La base de datos se puede implementar en un tipo de servicio IAAS o PAAS. En un IAAS se debería instalar el sistema operativo, el sistema gestor de base de datos y el administrador de base de datos. Mediante un tipo de servicio PAAS se puede escoger una base de datos exclusiva haciendo referencia a los siguientes proveedores; AWS, GCP y CNT. Los proveedores disponen de grandes soluciones y de diferentes productos, en la Tabla 1.8 se presenta

una comparación de las diferentes bases de datos tipo PAAS que dispone cada proveedor.

Tabla 1.8. Comparación de bases de datos de tres proveedores de *Cloud Computing*.

Proveedor	Amazon Web Services	Google Cloud Platform	CNT
Base de datos relacional	Compatible con MySQL y PostgreSQL, proporciona capacidad rentable y de tamaño modificable. Amazon Aurora, Amazon RDS	Servicio de base de datos MySQL y PostgreSQL totalmente administrado, frameworks web, datos estructurados, cargas de trabajo de procesamiento de transacciones online. Google Cloud SQL	Ofrece la posibilidad de instalar base de datos MySQL sobre una máquina virtual. No hay productos específicos para base de datos.
Base de datos NoSQL	Sin servidor, ofrece latencia de milisegundos, totalmente administrada en la nube. Amazon DynamoDB	Base de datos de columnas anchas NoSQL escalable y totalmente administrada, apta para cargas de trabajo de análisis y acceso en tiempo real. Google Cloud Bigtable	Manejo sobre máquinas virtuales. No hay productos específicos para base de datos.
Base de datos de gráficos	Es un producto de base de datos de gráficos totalmente administrado por Amazon. Amazon Neptune	Almacenamiento y transmisión de datos multimedia, muy fiable y económico. Google Cloud Storage	Manejo sobre máquinas virtuales con capacidades de procesamiento y almacenamiento. No hay productos específicos para base de datos.

En un IAAS al tener solo la infraestructura, permite instalar todo lo necesario de acuerdo a los requerimientos y así acoplar el sistema para que cualquier servicio pueda ser alcanzado a la base de datos y en un PAAS como se señaló en la Tabla 1.8 se tiene diferentes proveedores donde la base de datos viene prefabricada.

1.3.4 Descripción de Herramientas y Tecnologías Utilizadas

Para el desarrollo del prototipo de gestión médica es necesario el uso de varias herramientas y lenguajes. A continuación, se hará una descripción de las herramientas y lenguajes que se usarán en la elaboración de este Proyecto de Titulación.

a. Servicios Web JSON

Un servicio web (*web service*) es una colección de protocolos abiertos y estándares usados que pueden ser consumidos a través de la red. Los servicios web representan una forma estandarizada de integrar aplicaciones sin importar el lenguaje de programación o tecnología usada para su desarrollo, es decir, se puede integrar una aplicación desarrollada en PHP que se comunique con una aplicación desarrollada en Java o .Net, esto mediante el uso de XML, SOAP, JSON, que son tecnologías para la transferencia de datos. En la actualidad el uso de servicios web para Android se ha hecho más común, ya que con más frecuencia se usa la comunicación y la transferencia de datos entre el dispositivo móvil y los servidores, de los cuales se obtiene información[28] [29].

A continuación, se describirá el servicio web JSON, que será el utilizado en el desarrollo de este proyecto por sus ventajas a citar en el siguiente apartado.

JSON (*JavaScript Object Notation*) es un formato ligero basado en texto estándar para representar datos estructurados. Es comúnmente utilizado para intercambio de datos en aplicaciones web. JSON es un formato completamente independiente del lenguaje pero utiliza convenciones de lenguajes conocidos como C, C++, C#, Java, PHP [30].

La Figura 1.14 muestra la arquitectura que seguirá un sistema haciendo uso del servicio web; uno o varios clientes se comunican a un servidor remoto para realizar peticiones, las mismas que se realizan a través de un servicio web, y este a la vez devuelve el resultado de dicha petición. El servicio web será desarrollado en PHP y enviará la información en formato JSON para que la aplicación pueda entenderla.

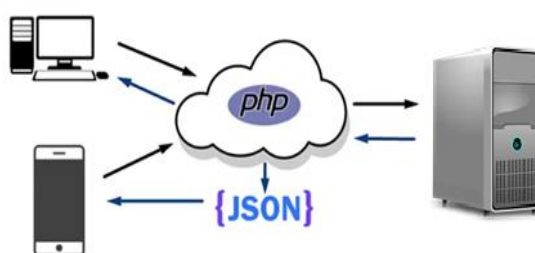


Figura 1.14. Arquitectura de un sistema utilizando un servicio web en la nube.

b. IDE Android Studio

Android Studio es un Entorno de Desarrollo Integrado (IDE por sus siglas en inglés) [31], basado en IntelliJIDEA⁷. Android Studio reemplazó a Eclipse como el IDE oficial para la plataforma Android. Es desarrollado en JAVA y está disponible para plataformas como Microsoft Windows, MacOS y GNU/Linux [31].

Android Studio es un IDE completo que ofrece funciones que aumentan la productividad durante la compilación de aplicaciones para Android. Entre las más importantes se puede mencionar:

- Un sistema de compilación basado en Gradle⁸ flexible.
- Un emulador rápido con varias funciones.
- Un entorno unificado que permite el desarrollo para todos los dispositivos Android
- Gran cantidad de herramientas y *frameworks* de prueba.

Cada proyecto realizado en este IDE está constituido por uno o más módulos que deben contener archivos de código fuente y archivos de recursos. Cada proyecto dispone de los siguientes módulos:

- Módulos de apps para Android.
- Módulos de bibliotecas.
- Módulos de Google App Engine.

La Figura 1.15 muestra los archivos que se crean por defecto en cada proyecto; en los mismos se guardarán todos los archivos que se crearán durante el desarrollo del mismo, pudiendo ser: clases, imágenes, etc.

Android Studio presenta una interfaz amigable para el desarrollo de aplicaciones nativas para Android. La Figura 1.16. muestra la ventana principal la cual consta de varias áreas lógicas. Es así que en esta interfaz se puede encontrar: la barra de herramientas, la barra de navegación, la ventana del editor y la barra de estado.

⁷IntelliJIDEA es un ambiente de desarrollo integrado para el desarrollo de programas informáticos

⁸Gradle es un sistema de compilación que reúne en uno solo las mejores prestaciones de otros sistemas de compilación, permitiendo así la reutilización de código.

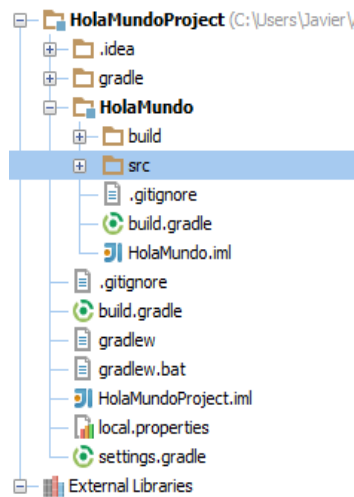


Figura 1.15. Archivos de un proyecto en la vista de Android.

El IDE de Android Studio dispone de todas las herramientas para tener un mejor desarrollo en una aplicación, además de componentes gráficos útiles y llamativos. Este IDE permite tener una compilación rápida y una ejecución de la aplicación en tiempo real, dispone de un emulador integrado o a su vez la posibilidad de ejecutar la aplicación directamente en el móvil.

c. IDE NetBeans

El IDE de NetBeans es un entorno de desarrollo visual de código abierto para aplicaciones programadas mediante Java. Sin embargo NetBeans también permite crear aplicaciones web con PHP. Los servicios web que se implementarán para la comunicación de la aplicación con el servidor serán desarrollados con este IDE [32].

d. WampServer

WampServer es el acrónimo usado para describir un sistema de infraestructura de Internet que usa las siguientes herramientas: Windows como Sistema Operativo, Apache como servidor web, MySQL como gestor de base de datos y PHP como lenguaje de programación [33]. El uso de WampServer permite generar páginas HTML, sistemas de base de datos y además gestionar servicios web para la comunicación de aplicaciones. En sí WampServer es un programa que permite que el computador se convierta en un servidor web. A continuación, se hará una descripción de cada uno de sus componentes.

- **Windows:** Sistema operativo desarrollado por Microsoft Windows, es el nombre de una familia de distribuciones de software para computadores, *smartphones* y servidores.

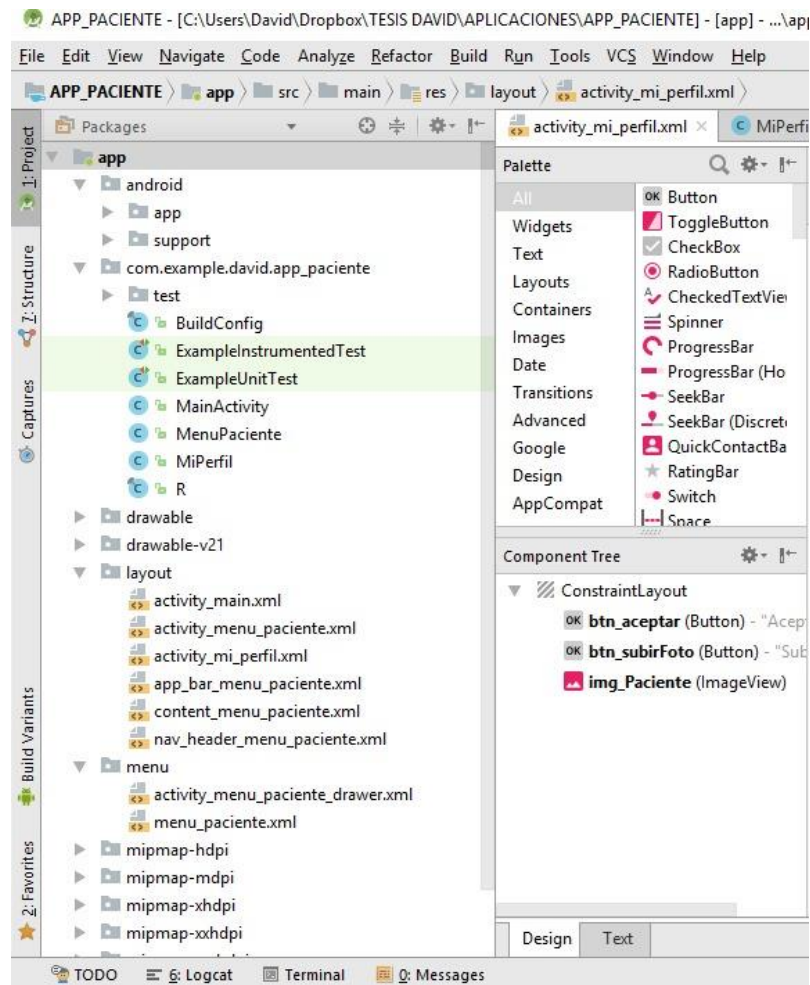


Figura 1.16. Ventana principal de Android Studio.

- **Apache:** Servidor Web HTTP de código abierto, que es utilizado para plataformas Linux, Microsoft Windows y Macintosh.
- **MySQL:** Sistema de gestión de bases datos relacional la cual es administrada en el caso de WampServer por *phpMyAdmim*.
- **PHP:** (*Hypertext Preprocessor*), es un lenguaje de programación de código abierto muy popular especialmente adecuado para el desarrollo web.

1.3.5 Metodología Ágil de Desarrollo Kanban

Originalmente la metodología Kanban surgió de la industria de manufactura de Toyota para la gestión de proyectos, no obstante en los últimos años no solo se ha utilizado en la industria sino también en el desarrollo de software [34]. Kanban es una palabra de origen japonés que significa “tarjeta”. El método de organización que utiliza es tipo *pull*, esto quiere decir que el equipo decide cuándo y cuánto trabajo se compromete a realizar, y solo podrán recibir nuevo trabajo cuando el anterior haya sido terminado [35].

“Kanban utiliza un mecanismo de control visual para hacer seguimiento del trabajo conforme este viaja a través de un flujo de valor. Para esto se usa un tablero con notas adhesivas o un panel electrónico con tarjetas” [35]. Kanban utiliza 3 principales reglas:

- Visualizar el flujo de trabajo.
- Limitar el trabajo en curso (WIP, WORK IN PROGRESS).
- Medir el tiempo en completar una tarea.

a. Flujo de Trabajo Kanban

Kanban propone un desarrollo incremental, por este motivo el primer paso será visualizar todo el trabajo y que esté dividido en partes (tareas). Como se mencionó cada una de estas tareas deben ser representadas en una nota etiquetada la cual va a contener información de la tarea y el tiempo estimado para que sea desarrollada. Para realizar un desarrollo ágil, es una práctica común e importante el poder visualizar y compartir el estado de un proyecto mediante tarjetas pegadas en un tablero. Cada una de estas tarjetas pegadas que básicamente representan una tarea en dicho tablero se las conoce como Tarea Kanban o Software Kanban. Este tablero está dividido en columnas, cada una de éstas con una fase de ciclo de producción (flujo de trabajo). En la Figura 1.17 se presenta un ejemplo de una representación de un tablero Kanban, la cual está compuesta de 3 fases:

1. Lista de Tareas: Se muestra la lista de tareas a realizar.
2. Tareas en Proceso: Se muestra la lista de tareas en desarrollo.
3. Tareas Realizadas: Se muestra la lista de tareas se encuentran terminadas.




Lista de Tareas	Tareas en Proceso	Tareas Realizadas
		

Figura 1.17. Ejemplo de un tablero con tarjetas de trabajo Kanban.

b. Trabajo en Proceso (WIP, WORK IN PROGRESS)

Kanban define un número límite de tareas a realizar por cada fase, esto es reconocido como limitar el trabajo en progreso (*work in progress*), con el objetivo de que nuevas tareas no sean ingresadas a cada fase mientras no se hayan terminado las anteriores. Esto para evitar los conocidos cuellos de botella en Kanban [35] [36].

En resumen, se puede decir que Kanban es una metodología donde su valor reside en su simplicidad para motivar a los desarrolladores a observar y controlar las actividades del proyecto.

2. METODOLOGÍA

El presente capítulo se divide en dos apartados: el **Diseño** y la **Implementación**. En la fase de Diseño se definirán los requerimientos que debe cumplir el prototipo, los cuales establecerán el comportamiento del sistema. A continuación, se tiene la implementación del prototipo, donde se encuentra la instalación de herramientas y la codificación del diseño. Tanto para la etapa del diseño como para la implementación los componentes que integran el prototipo serán separados en 3 Capas: Datos, Negocio y Presentación.

2.1 Diseño

En este apartado, se detalla el diseño a realizar para el prototipo del sistema móvil para la gestión médica del área de traumatología para el “Club de Leones Quito”. Como punto inicial según la metodología Kanban, se debe definir un tablero de actividades que servirá como guía para todo este desarrollo. Como siguiente paso, se definirán los requerimientos tanto funcionales como no funcionales del sistema, desde la perspectiva de médico y paciente. Con estos requerimientos se realizarán los respectivos diagramas de casos de uso. A continuación, se realizará la selección del proveedor que se usará como plataforma en la nube para posteriormente realizar el diseño de la capa de Base de Datos y de la capa de Negocio en donde se definirán todos los servicios de software. Finalmente se diseñará la capa de Presentación.

2.1.1 Tablero de Actividades Kanban

TABLERO KANBAN “CLUB DE LEONES QUITO”			
Lista de Tareas	Tareas en Proceso	Horas	Tareas Realizadas
Creación de una instancia EC2 en AWS.	Realizar encuestas a médicos del área de Traumatología del “Club de Leones Quito”.	3	
Instalación y codificación de WampServer.	Identificación de requerimientos funcionales (RF).	7	
Implementación de la Capa de Datos.	Identificación de requerimientos no funcionales (RNF).	10	
Implementación de la Capa de Negocio.	Selección del Proveedor de Cloud Computing	10	
Codificación de las Clases.	Clasificación de los RF por módulos.		
Codificación de los servicios Web.	Diseño de los diagramas de casos de uso.	7	
Implementación de la Capa de Presentación.	Diseño del diagrama relacional de base de datos.		
Pruebas de funcionamiento al módulo de registro de usuarios (médico y paciente).	Diseño del diagrama de clases.	10	
Pruebas de funcionamiento del módulo de historiales clínicos.	Diseño de los servicios Web.	10	
Pruebas de funcionamiento del módulo de registro y visualización de sugerencias de visitas médicas.	Diseño de la capa de presentación.	10	
Pruebas de funcionamiento del módulo de registro de sugerencias por parte de los pacientes.	Diseño de las vistas de la aplicación móvil.	10	
Pruebas de funcionamiento del módulo de carga de exámenes para el usuario paciente.			
Pruebas de integración.			
Presentar resultados.			

Figura 2.1. Tablero de Actividades Kanban del Diseño.

La Figura 2.1 muestra el Tablero de actividades Kanban el cual permitirá organizar las tareas de diseño del Sistema.

En la Figura 2.1 se presentan las siguientes columnas: “*lista de tareas*” que son todas las actividades pendientes por realizar en las etapas de implementación y pruebas; la columna de “*Tareas en proceso*” contiene las actividades que se encuentran en etapa de desarrollo, y en este apartado corresponde a la fase de diseño del prototipo; y la columna “*Tareas Realizadas*” corresponden a la lista de actividades completadas, que al momento todavía no se ha realizado ninguna.

2.1.2 Análisis de Requerimientos

Los requerimientos del prototipo serán definidos con base a las encuestas realizadas, las mismas que se encuentran en el **ANEXO I**. Éstas fueron realizadas a los médicos del área de Traumatología del “Club de Leones Quito” cuyos resultados se encuentran en el mismo anexo.

a. Requerimientos Funcionales (RF)

Según las encuestas realizadas y en coordinación con el plan aprobado, se definen los RF los cuales indican las funciones del sistema de software. La Tabla 2.1 describe los RF que tendrá el sistema para el usuario Médico (RFM).

Tabla 2.1. Descripción de los requerimientos funcionales del médico.

RF	DESCRIPCIÓN
RFM01	Registrar médico.
RFM02	Loguear médico.
RFM03	Visualizar información personal médico.
RFM04	Modificar información personal médico.
RFM05	Agregar paciente.
RFM06	Modificar paciente.
RFM07	Eliminar paciente.
RFM08	Buscar paciente.
RFM09	Visualización de pacientes.
RFM10	Agregar sugerencia de visita.
RFM11	Visualizar agenda.
RFM12	Visualizar sugerencias.
RFM13	Cargar exámenes médicos.
RFM14	Eliminar exámenes médicos.
RFM15	Crear Historial.
RFM16	Crear diagnóstico.
RFM17	Solicitar exámenes médicos.
RFM18	Añadir descripción (estado de salud).
RFM19	Añadir receta.
RFM20	Generar Reporte.

La Tabla 2.2 describe los RF que tendrá el sistema para el usuario paciente (RFP).

Tabla 2.2. Descripción de los requerimientos funcionales del paciente.

RF	DESCRIPCIÓN
RFP01	Loguear paciente.
RFP02	Visualizar información personal.
RFP03	Visualizar sugerencia de visitas.
RFP04	Cargar exámenes médicos.
RFP05	Añadir sugerencias de la atención prestada.

b. Requerimientos No Funcionales (RNF)

Los RNF son aquellos que restringen las características del software [37]; por ejemplo, la disponibilidad, integridad, tiempo de acceso al sistema, etc. Estos requisitos proveerán las características en el comportamiento del software completando las funcionalidades. La Tabla 2.3 muestra los requerimientos no funcionales y su respectiva descripción.

Tabla 2.3. Descripción de los requerimientos no funcionales.

ETIQUETA	RNF	DESCRIPCIÓN
Presentación	RNF01	Los usuarios deben contar con una interfaz que sea amigable e intuitiva.
Disponibilidad	RNF02	El sistema deberá estar disponible las 24 horas.
Integridad	RNF03	Los datos no deberán ser corruptos por transacciones incompletas o fallidas.
Tiempo de respuesta	RNF04	El tiempo de respuesta deberá ser de pocos segundos. Esto dependerá mucho de la velocidad de acceso a Internet.
Escalabilidad	RNF05	El prototipo debe ser construido en una plataforma que permita el incremento en recursos como almacenamiento, memoria RAM.
Capacidad	RNF06	El sistema deberá ser capaz de soportar al menos 7 usuarios simultáneos.

2.1.3 Selección del Proveedor de Cloud Computing

El proveedor de *Cloud Computing* para este Proyecto de Titulación, en base a lo analizado en el **Capítulo 1 Apartado 1.3.3**, será un proveedor de servicio de tipo IAAS ya que presenta otras características diferentes a un tipo PAAS como por ejemplo: elección del sistema operativo, capacidad almacenamiento y memoria, manejo de herramientas de acuerdo a la necesidad del proyecto.

Esta decisión también se ha visto respaldada en base a el cuadrante mágico de Gartner [38]. El cuadrante mágico de Gartner proporciona a las empresas un medio para identificar y diferenciar a los proveedores de servicios del sector de las Tecnologías de la

Información. Según define Gartner, los principales fabricantes de software que operan bien hoy en día están liderando los cuadrantes mágicos, ya que tienen una visión clara de la dirección del mercado y siempre se mantienen en innovación para así mantener su posición de líderes en el mercado [38].

Los criterios de evaluación que tiene los consultores de Gartner se basan en los siguientes factores [39]:

- Análisis del producto en su capacidad para ejecutar el servicio del mismo.
- Usabilidad, rapidez, y su viabilidad en el tiempo.
- La cantidad de clientes que utilizan la solución y las recomendaciones.
- Escalabilidad, accesibilidad, rendimiento, y seguridad

En el informe 2016 *Magic Quadrant for Cloud Infrastructure as a Service, Worldwide* [40], Gartner situó a AWS entre los líderes del cuadrante por sexto año consecutivo, ya que posee la visión más completa y la mayor capacidad de ejecución [40]. La Figura 2.2 muestra el cuadrante mágico donde están los principales proveedores de *Cloud Computing*.



Figura 2.2. Cuadrante Mágico de Gartner [40].

Por todo lo citado anteriormente AWS, ha sido escogido como proveedor para el desarrollo de este Proyecto de Titulación. En la Tabla 2.4 se muestran las características principales de AWS.

Tabla 2.4. Características del Proveedor AWS [20].

Facilidad de uso	AWS está diseñado para permitir que los proveedores de software independientes y proveedores de aplicaciones puedan hospedar de una forma rápida y segura su aplicación.
Flexible	AWS permite seleccionar el sistema operativo, el lenguaje de programación, la plataforma de aplicaciones web, base de datos. Con AWS, tendrá acceso a un entorno virtual que le permite cargar el software y los servicios que necesite la aplicación.
Rentable	Únicamente tendrá que afrontar el costo de la potencia de cómputo, el almacenamiento y demás tipos de recursos que utilice.
De confianza	Con AWS, tendrá a su disposición una infraestructura, escalable, segura y de confianza.
Escalabilidad y alto desempeño	Las herramientas AWS como Auto Scaling y Elastic Load Balancing, su aplicación podrá ampliarse o reducirse según la demanda.

En el diseño del servidor que se utilizará en la nube, AWS ofrece una imagen de Máquina Virtual (MV). Esta MV tendrá como sistema operativo Windows Server 2012. La Tabla 2.5 muestra ciertas características del servidor que se utilizará, tomando en cuenta que para el presente prototipo se ha escogido una instancia de nivel libre lo que significa “cero costos por un año”, siempre que no se exceda los límites establecidos.

Tabla 2.5. Características del Servidor AWS.

Sistema Operativo	Windows Server 2012 R2 Base
Modelo	T2.micro
CPU virtual	1
Memoria (GB)	1
Tiempo de cómputo disponible al mes	750 horas
Almacenamiento (GB)	20

2.1.4 Diseño de la Capa de Datos

En este apartado se empezará realizando un diseño del diagrama entidad relación en donde están las tablas (entidad) con sus pertinentes campos y el cómo estas tablas se relacionan unas con otras por medio de campos en común (relaciones). Se hace notar que cada relación tiene una cardinalidad definida, es decir, el número de veces que una entidad aparece asociada a otra.

a. Diagrama Entidad Relación

En la Figura 2.3 se describe el Diagrama Entidad-Relación que tendrá el Sistema de Gestión Médica. Este diagrama, es un tipo de esquema de flujo que ilustra cómo las entidades, como personas, objetos o conceptos se relacionan entre sí dentro de un sistema [41]. En el diagrama se indica la relación que existen entre las tablas *Medico*, *Diagnostico* y *Paciente*. La tabla *Diagnostico* permite guardar los diagnósticos creados por un médico a un determinado paciente, esta relación permite saber qué médico fue el que atendió al paciente. En cada visita el médico podrá solicitar exámenes médicos los cuales serán guardados en una tabla *ExamenMedico*, esta tabla tiene una relación con las tablas *Paciente* y *Medico* permitiendo saber a qué paciente le pertenece el examen y que médico extendió la orden.

A continuación, se tiene una tabla *SugerenciaCita* la cual permite almacenar la sugerencia de la próxima visita que puede dar un médico y está anexada a la tabla *Paciente* y *Medico* permitiendo saber a qué paciente se le dará la respectiva sugerencia.

Finalmente, se tiene una tabla *SugerenciaPaciente* la cual permite almacenar las sugerencias de los pacientes acerca de la atención prestada, esta tabla tiene relación con la tabla *Paciente* y la tabla *Medico* permitiendo así que el paciente pueda dar la sugerencia o queja del médico que le atendió. Se debe mencionar que la diferencia entre la tabla *SugerirCita* y *SugerenciaPaciente*, es que la primera guardará la sugerencia de una próxima visita médica y la segunda almacenará comentarios acerca de la atención.

Las tablas *Medico* y *Paciente* tiene características (campos) específicas que son: identificador (id), nombres, username, contraseña, correo, teléfono. Las demás tablas tienen atributos de acuerdo a las necesidades de cada una, las cuales están desglosadas en el **ANEXO II**.

El diagrama fue generado utilizando el software *Lucidchart* [42] que es una herramienta *online* que permite a los usuarios el desarrollo y generación de flujo, organigramas, y/o diagramas Entidad-Relación.

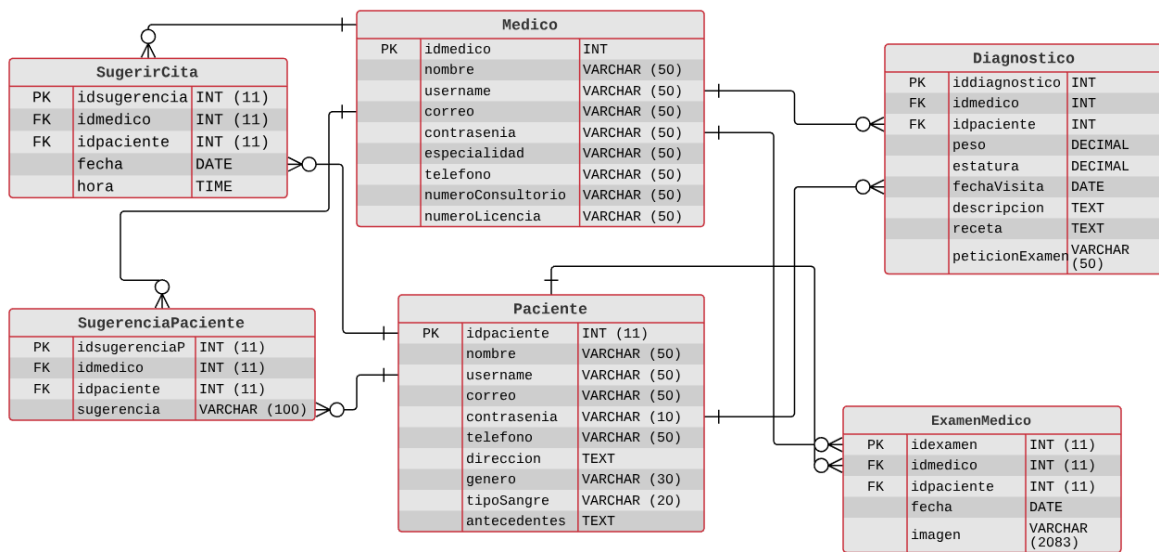


Figura 2.3. Diagrama Entidad-Relación de la capa de base de datos.

2.1.5 Diseño de la Capa de Negocio

Esta capa presenta la lógica de los procesos que permitirán implementar el Sistema de Gestión Médica, entre los cuales están los diagramas de casos de uso, el diagrama de clases y el diseño del servicio web.

a. Diagramas de Casos de Uso (DCU)

Los diagramas de caso de uso sirven para especificar la comunicación y el comportamiento del sistema y su interacción con los usuarios (actores). A continuación, se definen los actores los cuales van a interactuar con el Sistema de Gestión Médica. La Figura 2.4 muestra a los dos usuarios principales que son: el médico y el paciente.



Figura 2.4. Actores que interactúan con el sistema.

Con el fin de ordenar adecuadamente los RF, se ha decidido utilizar subsistemas y agruparlos de manera coherente.

La Figura 2.5 muestra todos los subsistemas y el nombre de los mismos con los cuales deberán interactuar los actores antes descritos, cada uno agrupa 1 o más RF, asegurando que se cumplan todas las funcionalidades necesarias.

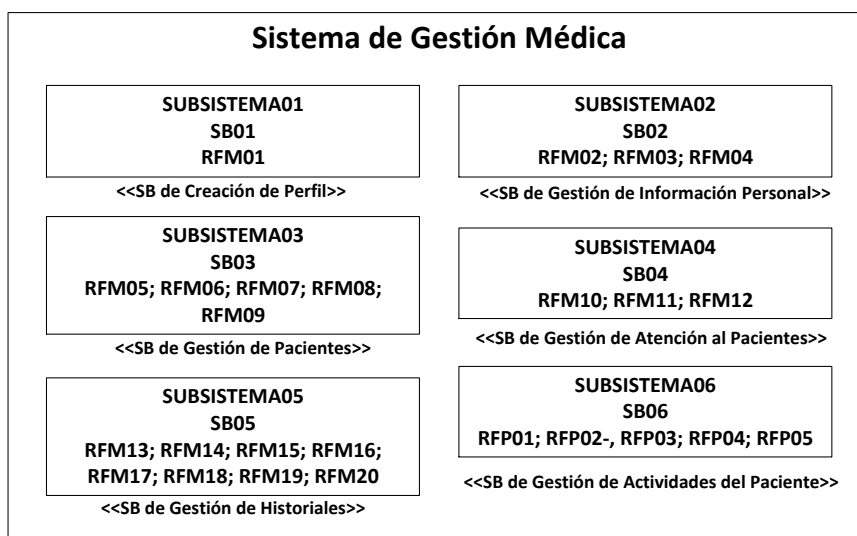


Figura 2.5. Subsistemas que conforman el Sistema de Gestión Médica.

Cada uno de los RF definidos en el **Apartado 2.1.2** deben estar relacionados a un subsistema, una vez relacionados se detallará los casos de uso y sus respectivos diagramas. En el presente Proyecto de Titulación se han identificado 5 casos de uso para el usuario Médico y 1 caso de uso para el usuario Paciente, los cuales asociados a los RF representan la funcionalidad del Sistema de Gestión Médica. A continuación, se mostrarán en la Tabla 2.6 los casos de uso identificados, los cuales también serán representados en el respectivo diagrama de casos de uso.

Tabla 2.6. Descripción de los RF del médico, el subsistema asociado, y los casos de uso.

SUBSISTEMA	CASO DE USO	RF	DESCRIPCIÓN
SB01	CU01	RFM01	Registrar médico.
SB02	CU02	RFM02	Loguear médico.
SB02	CU02	RFM03	Visualizar información personal médico.
SB02	CU02	RFM04	Modificar información personal médico.
SB03	CU03	RFM05	Agregar paciente.
SB03	CU03	RFM06	Modificar paciente.
SB03	CU03	RFM07	Eliminar paciente.
SB03	CU03	RFM08	Buscar paciente.
SB03	CU03	RFM09	Visualización de los pacientes.
SB04	CU04	RFM10	Agregar sugerencia de visita.
SB04	CU04	RFM11	Visualizar agenda.
SB04	CU04	RFM12	Visualizar sugerencias de la atención.
SB05	CU05	RFM13	Cargar exámenes médicos.
SB05	CU05	RFM14	Eliminar exámenes médicos.
SB05	CU05	RFM15	Crear Historial.
SB05	CU05	RFM16	Crear diagnóstico.
SB05	CU05	RFM17	Solicitar exámenes médicos.
SB05	CU05	RFM18	Añadir descripción (estado de salud).
SB05	CU05	RFM19	Añadir receta.
SB05	CU05	RFM20	Generar Reporte.

La Tabla 2.6 muestra también la relación que existe entre el subsistema con cada RF y los respectivos casos de uso.

La Tabla 2.7 describe los RF que tendrá el sistema para el usuario Médico, el subsistema con los cuales están relacionados y los respectivos casos de uso.

Tabla 2.7. Descripción de los RF del paciente, el subsistema asociado y el caso de uso.

SUBSISTEMA	CASO DE USO	RF	DESCRIPCIÓN
SB06	CU06	RFP01	Logear paciente
SB06	CU06	RFP02	Visualizar información personal
SB06	CU06	RFP03	Visualizar sugerencia de visitas
SB06	CU06	RFP04	Cargar exámenes médicos
SB06	CU06	RFP05	Añadir sugerencias de la atención prestada

El Subsistema **Creación de Perfil (SB01)** que se muestra en la Figura 2.6 es el encargado de que los médicos puedan registrarse en el aplicativo para que puedan utilizar el sistema.

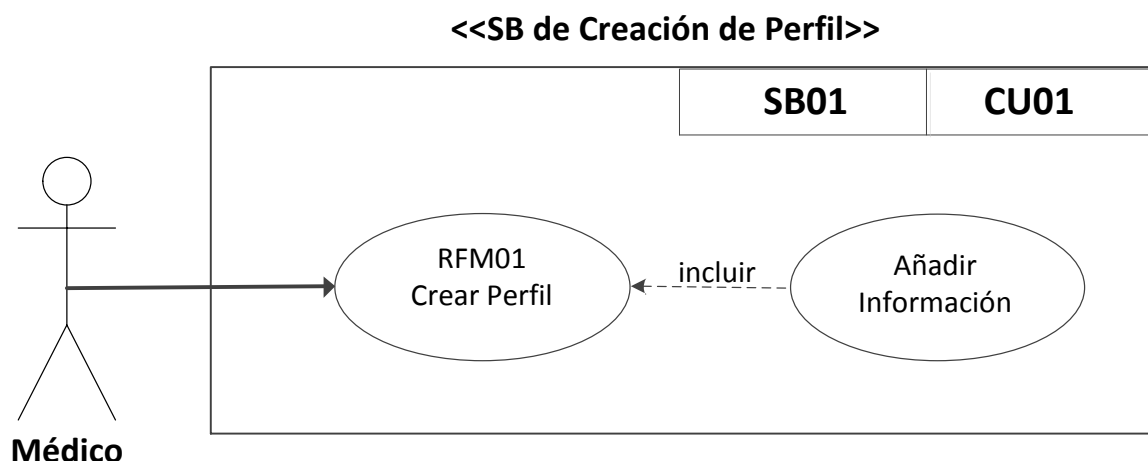


Figura 2.6 Caso de uso registro de médico.

La creación de perfil solo lo podrá realizar un usuario de tipo médico. El usuario de tipo paciente tendrá acceso al sistema una vez que el médico lo haya registrado como un nuevo paciente.

El Subsistema de **Gestión de Información Personal (SB02)** que se muestra en la Figura 2.7 permite al médico visualizar y modificar su información personal. Esto lo podrá realizar una vez *logueado* en el sistema.

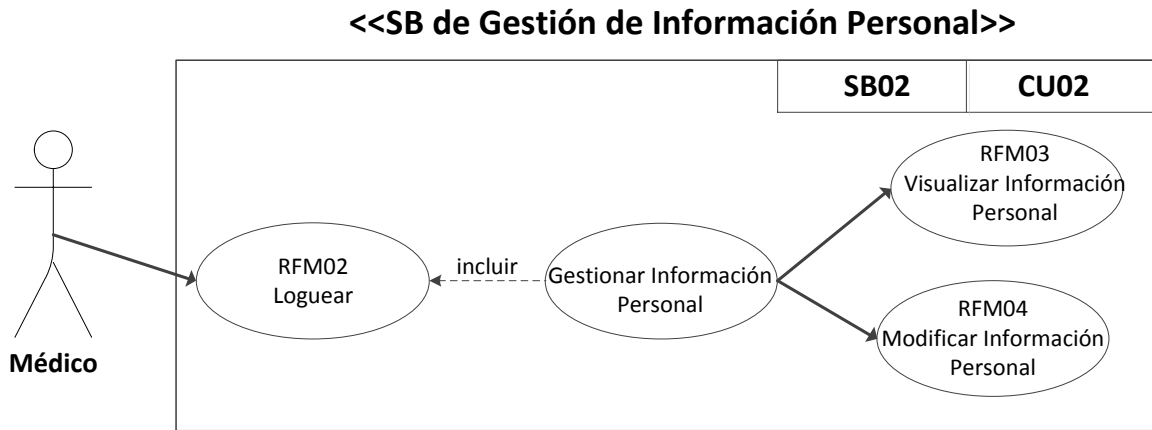


Figura 2.7. Caso de uso de gestión de información personal.

El Subsistema de **Gestión de Pacientes (SB03)** mostrado en la Figura 2.8 permite al médico: agregar, modificar, eliminar o buscar pacientes; esto como parte de la administración de pacientes.

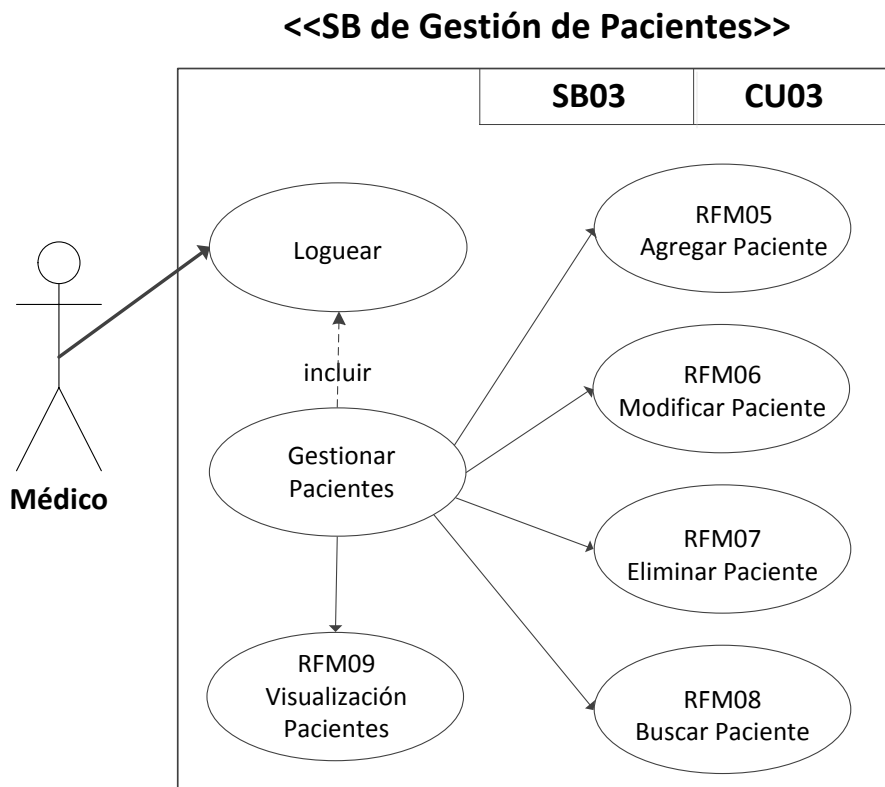


Figura 2.8. Caso de uso de gestión de pacientes.

El Subsistema de **Gestión Atención del Paciente (SB04)** se muestra en la Figura 2.9 y se encarga de agregar una sugerencia de visita, visualizar agenda y visualizar las sugerencias que puedan hacer los pacientes.

<<SB de Gestión de Atención al Paciente>>

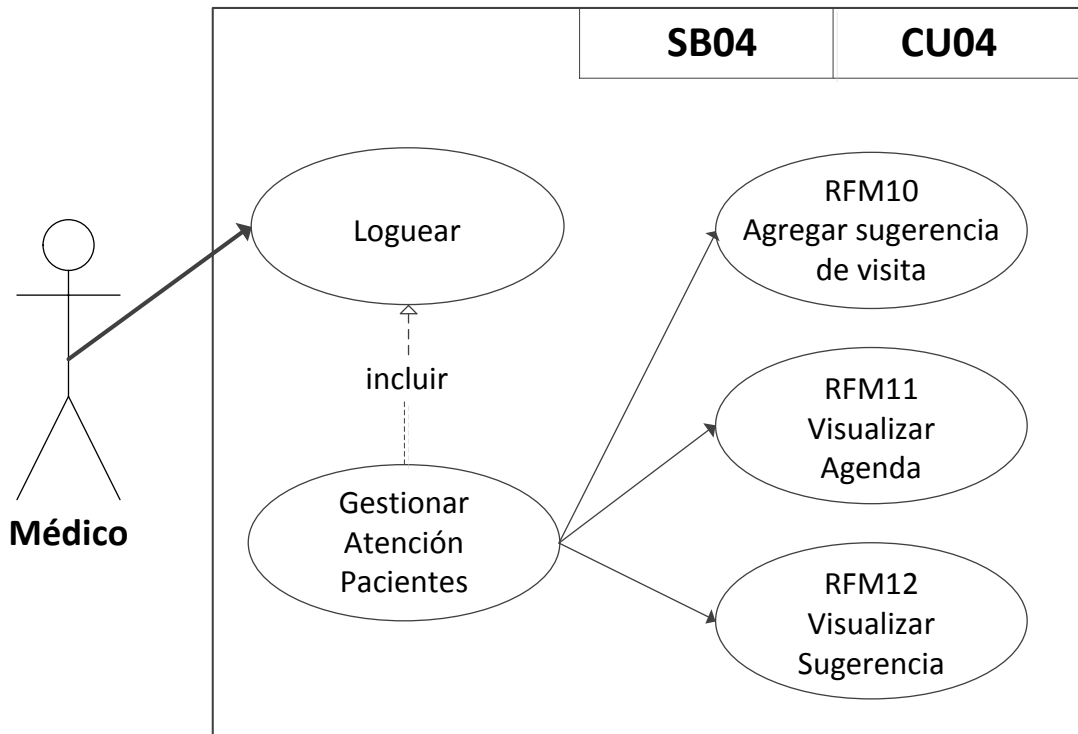


Figura 2.9. Caso de uso de gestión de atención de pacientes.

El Subsistema de **Gestión de Historiales (SB05)** indicado en la Figura 2.10 es el encargado de que el médico pueda crear los historiales del paciente.

<<SB de Gestión de Historiales>>

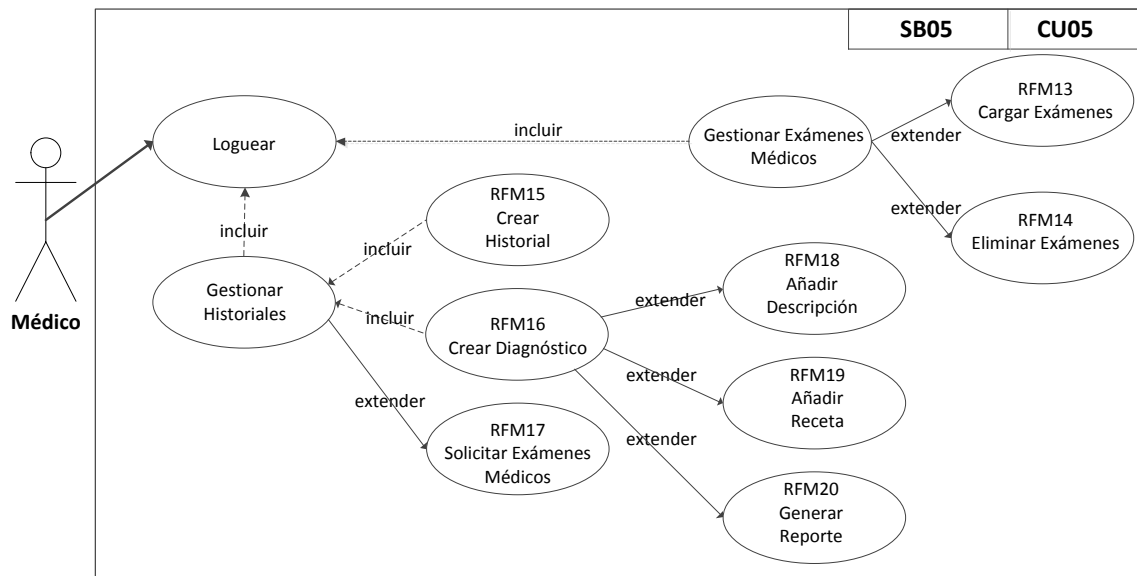


Figura 2.10. Casos de uso de gestión de historiales.

El SB05 permitirá crear los diagnósticos del paciente permitiendo almacenar los avances en el proceso de rehabilitación del mismo, solicitar exámenes médicos y además de tener la opción de generar un reporte de cada diagnóstico.

El Subsistema de **Gestión de Actividades del Paciente (SB06)** que se muestra en la Figura 2.11, es el encargado de que el paciente pueda visualizar su información personal, visualizar las sugerencias de visitas que previamente le asignó el médico, cargar los exámenes médicos solicitados y añadir sugerencias sobre la atención recibida.

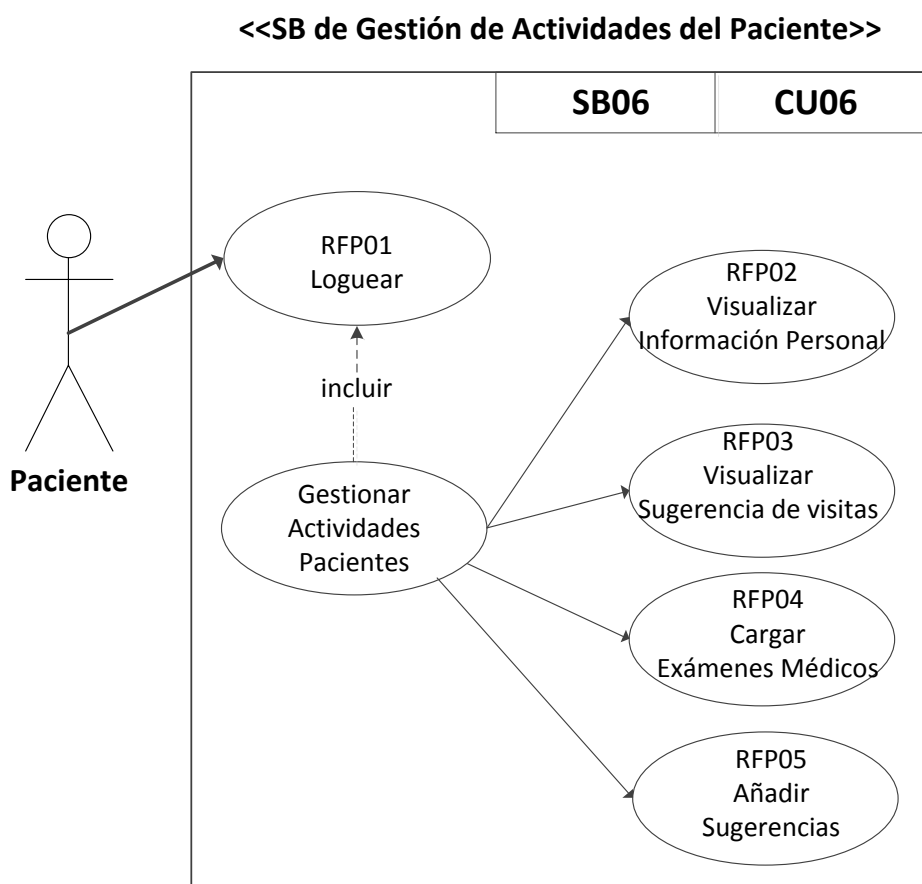


Figura 2.11. Caso de uso de gestión de actividades del paciente.

b. Diagrama de Clases

Un diagrama de clases presenta la estructura del sistema en un lenguaje Orientado a Objetos de manera gráfica. Las clases presentadas serán las que usará el sistema para todos los servicios que permitirán el intercambio de datos entre la capa de base de datos y la capa de presentación.

En la Figura 2.12 se muestra el diagrama de clases completo del Sistema de Gestión Médica. Aquí se muestran las clases principales y los métodos más importantes que definen el funcionamiento del Software de Gestión Médica.

El diagrama de clases al igual que el diagrama Entidad-Relación fue realizado mediante el software 'Lucidchart' [42].

Los atributos de las clases son de tipo privado, es así que para acceder a sus atributos y modificarlos es necesario definir métodos públicos 'get()' y 'set()', entregando así encapsulamiento. Se observa que la clase *Medico* y *Paciente* heredan atributos de una clase *Persona* al hacer uso de una relación de herencia. Además, se observa que las clases *Medico* y *Paciente* tienen cardinalidad de 'uno' a 'n' con las clases *Cita*, *Diagnostico* y *Examen*, lo cual implica que las instancias de las clases *Medico* y *Paciente* pueden tener varias instancias con las clases *Cita*, *Diagnostico* y *Examen*. La clase *Sugerencia* solo tiene una relación con la clase *Paciente*, y tiene una cardinalidad de 'uno' a 'n' lo que implica que un paciente puede dar muchas sugerencias o quejas de la atención prestada.

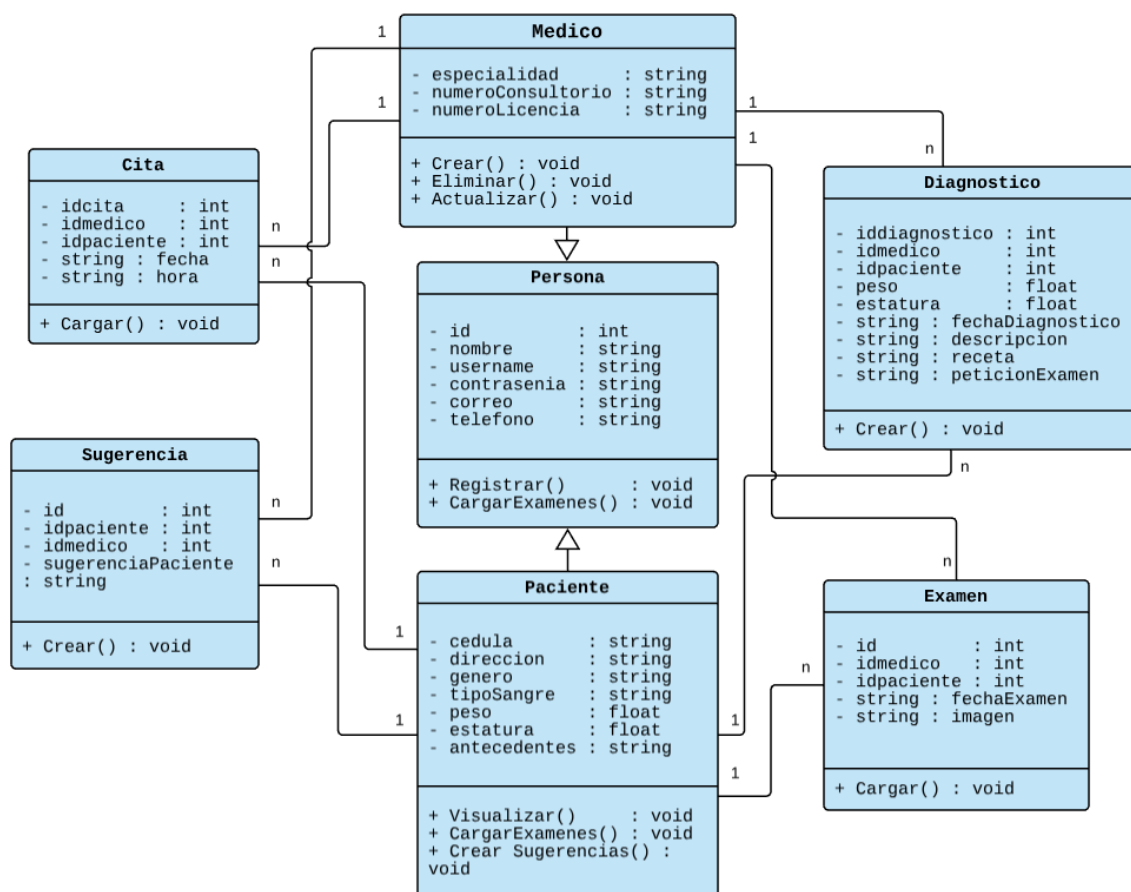


Figura 2.12. Diagrama de Clases.

c. Diseño de los Servicios Web

Para realizar los servicios web se utilizará el lenguaje de programación PHP; los cuales permiten darle varias funcionalidades al Sistema de Gestión Médica [43]. Las principales

ventajas de utilizar PHP para el desarrollo de los servicios web son: ser un lenguaje más simple que otros como C o JAVA. PHP se acopla a la mayoría de plataformas como Windows, Linux, Mac OS; está especialmente diseñado para soportar base de datos y servicios web [43]. La lógica de un servicio se puede mostrar en la Figura 2.13.

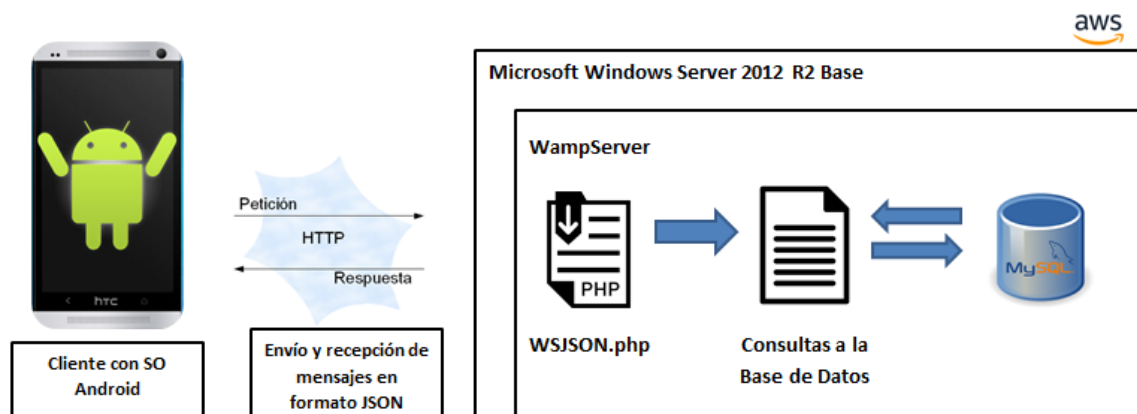


Figura 2.13. Arquitectura del Servicio Web.

La arquitectura que se muestra en la Figura 2.13 indica que un usuario con sistema operativo Android realizará peticiones al servidor mediante mensajes JSON; las peticiones serán concatenadas con el respectivo servicio web donde se podrá acceder a todos los beneficios y funcionalidades que presentan estos archivos codificados en PHP y a su vez a la base de datos MySQL. Así mismo el servidor devolverá respuestas de igual manera en formato JSON que deberán ser decodificadas en el lado del cliente para mostrar la información solicitada. El servicio web contiene la lógica para ejecutar la acción pedida por el cliente Android.

2.1.6 Diseño de la Capa de Presentación

En esta capa se muestra las interfaces que interactuarán con el usuario final. Estas interfaces corresponderán a la aplicación móvil, mismas que recolectan la información proporcionada por los usuarios Médico y Paciente.

El diseño de la Capa de Presentación se realizará en base a la herramienta llamada *PENCIL.PENCIL* [44] que es una herramienta que permite bosquejar las diferentes pantallas que puede tener una aplicación móvil.

a. Diseño de las Vistas de la Aplicación Médico

La aplicación contará con diferentes interfaces gráficas. A continuación, se presentan los diferentes bosquejos realizados para el usuario médico. La Figura 2.14 muestra el diseño de las interfaces del Subsistema **Creación de Perfil (SB01)**. Las interfaces han sido diseñadas tomándo en cuenta la opinión de los encuestados en forma y color, por

ejemplo puntos como: que sea intuitiva, contenga imágenes, el tamaño de los botones, la posición del menú, mensajes personalizados. También se hace notar que este diseño se ha planteado para cumplir con los requerimientos funcionales anteriormente definidos en el **Apartado 2.1.2.**

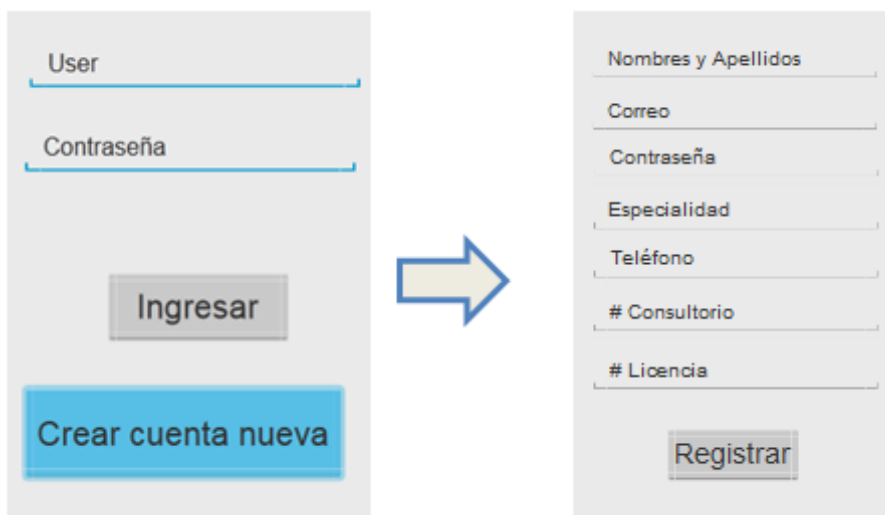


Figura 2.14. Interfaces de registro de médicos.

La Figura 2.15 muestra las interfaces del Subsistema de **Gestión de Información Personal (SB02)** necesarias para visualización del perfil médico, donde se podrá visualizar o modificar su información personal.



Figura 2.15. Pantallas de visualización del Perfil del usuario médico.

La Figura 2.16 y la Figura 2.17 muestran las interfaces del Subsistema de **Gestión de Pacientes (SB03)**. La Figura 2.16 muestra el registro de pacientes.



Figura 2.16. Interfaces de registro de pacientes.

A continuación, una vez que el médico haya registrado a un paciente, podrá visualizar a los pacientes mediante el botón de historiales clínicos, donde cada paciente tendrá diferentes opciones como se muestra en la Figura 2.17.

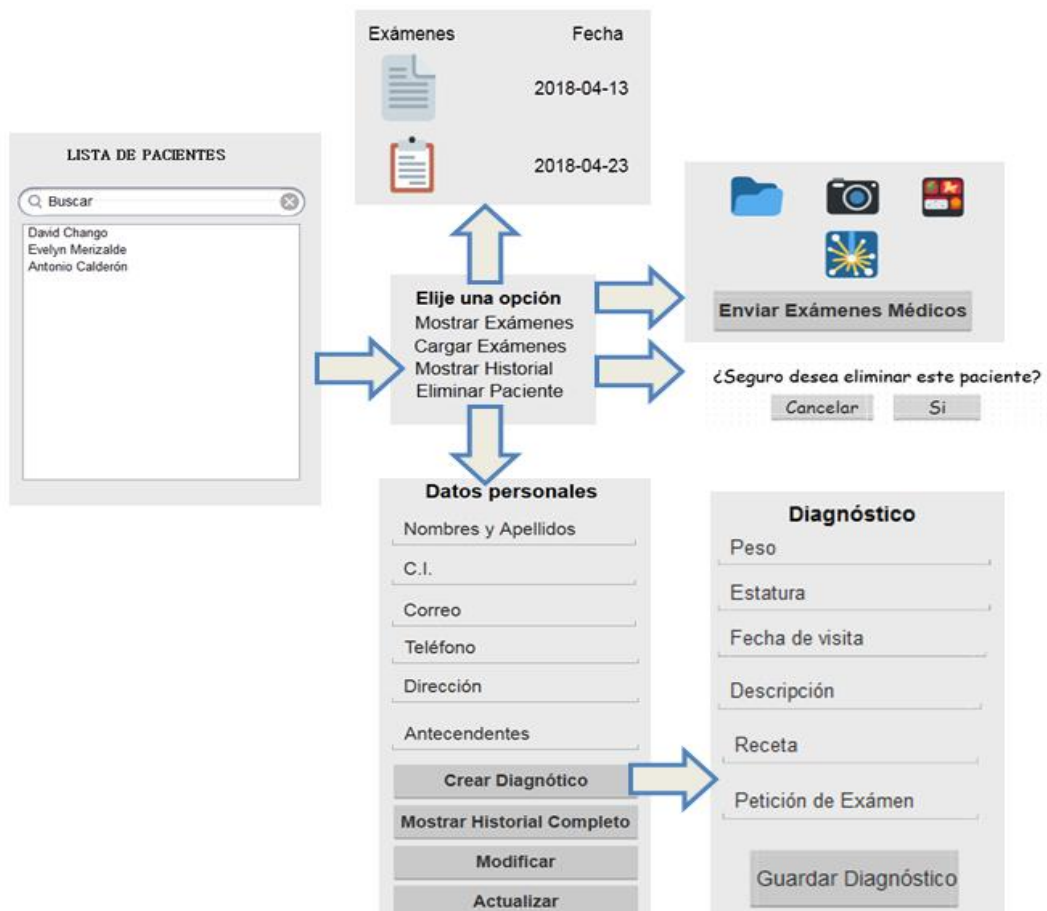
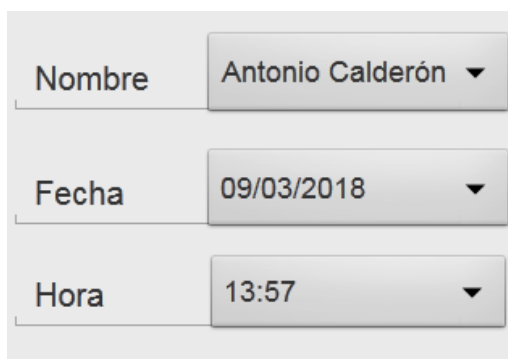


Figura 2.17. Interfaces de la gestión de información del paciente.

La Figura 2.17 muestra además las pantallas que el médico dispondrá para realizar toda la administración de la información de los pacientes; eliminar paciente, creación de diagnósticos, añadir los respectivos exámenes médicos y modificar la información personal del paciente.

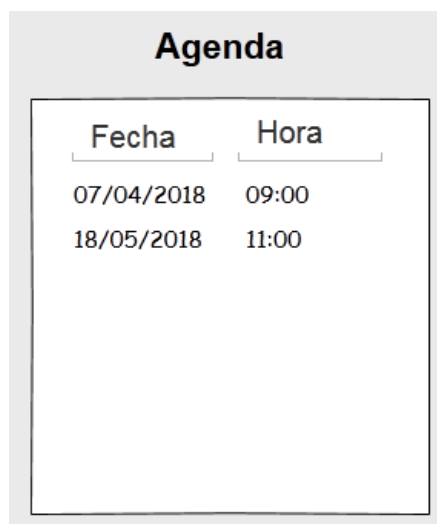
La Figura 2.18 y la Figura 2.19 muestran las interfaces del Subsistema de **Gestión de Atención al Paciente (SB04)**. La Figura 2.18 es la interfaz donde el médico podrá sugerir una cita médica a su paciente.



La imagen muestra una interfaz de usuario con tres campos de selección. El primer campo, etiquetado como 'Nombre', tiene un menú desplegable con 'Antonio Calderón' seleccionado. El segundo campo, etiquetado como 'Fecha', tiene un menú desplegable con '09/03/2018' seleccionado. El tercer campo, etiquetado como 'Hora', tiene un menú desplegable con '13:57' seleccionado.

Figura 2.18. Interfaz de sugerencia de cita médica.

La Figura 2.19 es la interfaz gráfica donde el médico podrá visualizar las citas médicas sugeridas.



La imagen muestra una interfaz de usuario con el título 'Agenda' en un encabezado. Debajo del título hay una tabla con dos columnas: 'Fecha' y 'Hora'. La tabla contiene dos filas de datos.

Fecha	Hora
07/04/2018	09:00
18/05/2018	11:00

Figura 2.19. Interfaz de visualización de sugerencia de citas médicas.

La Figura 2.20 muestra las interfaces del Subsistema de **Gestión de Historiales (SB05)**, que una vez que el médico haya creado diagnósticos, podrá visualizarlos todos a través del botón *Mostrar Historial Completo*. Cada diagnóstico podrá ser visualizado independiente y además ser guardado como documento en formato pdf.

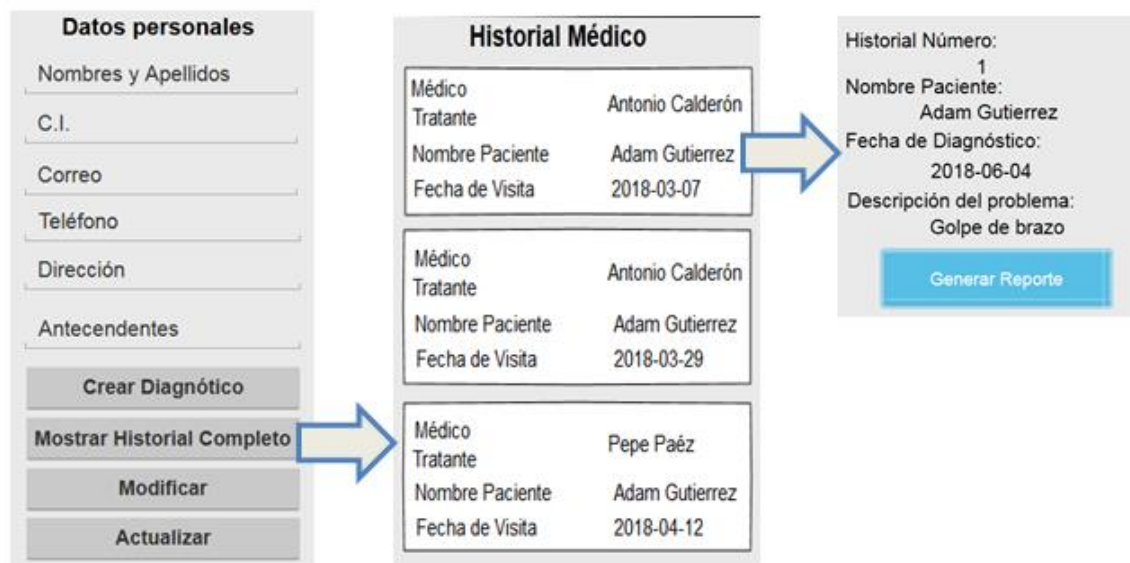


Figura 2.20. Interfaces de visualización para mostrar el historial completo.

b. Diseño de las Vistas de la Aplicación Paciente

La Figura 2.21 y la Figura 2.22 muestran las interfaces del Subsistema de **Gestión de Actividades del Paciente (SB06)**. La Figura 2.21 muestra la pantalla de *login* y las actividades que podrá realizar el paciente. Posteriormente, el paciente solo podrá acceder una vez que un médico lo haya registrado en el sistema. El acceso será con un nombre de usuario y una contraseña que para este usuario será su número de cédula.

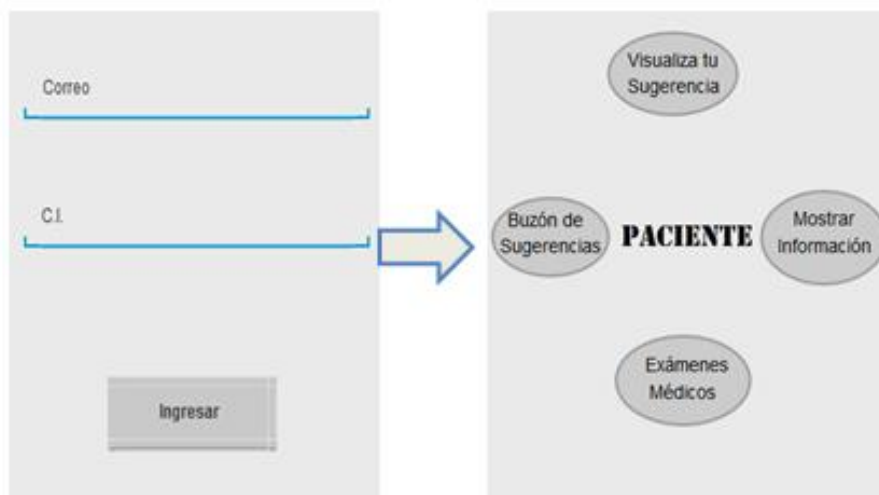


Figura 2.21. Interfaces de acceso al sistema usuario Paciente.

La Figura 2.22 muestra las diferentes actividades que podrá realizar el paciente, pudiendo ser: realizar una sugerencia de cómo fue la atención médica, visualizar la sugerencia de citas médicas, cargar al sistema sus respectivos exámenes médicos y visualizar su información personal.

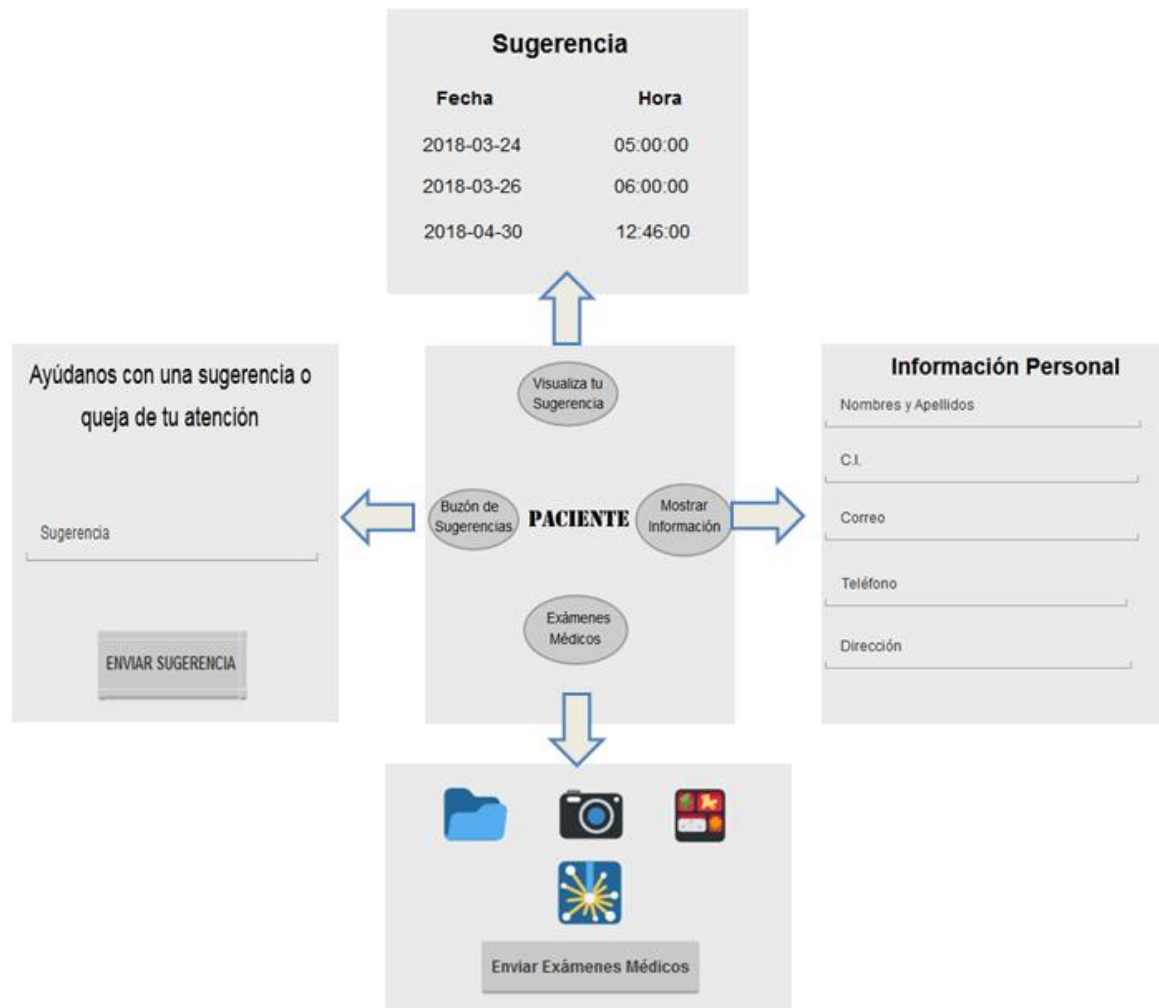


Figura 2.22. Diseño de interfaces del usuario paciente.

2.2 Implementación

A continuación, se muestra el proceso que se llevará a cabo para la implementación del sistema. En esta fase, en primera instancia se realizará la actualización del tablero Kanban con las actividades necesarias para esta etapa; se presenta también la instalación y codificación de las herramientas de software necesarias; y además de la codificación de todo lo necesario para el Sistema de Gestión Médica siguiendo el diseño planteado en el **Apartado 2.1**. Finalmente, se muestra un ejemplo de la codificación de las interfaces de la aplicación a desarrollar.

2.2.1 Tablero de Actividades Kanban

La Figura 2.23 muestra el tablero actualizado para continuar con la etapa de implementación. El tablero de Kanban ejemplifica el estado de las tareas, en la columna “*Tareas Realizadas*” se observa que las actividades para la etapa de diseño están

terminadas, y en la columna de “*Tareas en Proceso*” son aquellas que corresponden con la fase de implementación. Finalmente, la columna “*Lista de Tareas*” son las actividades que se realizarán en la fase de pruebas.

TABLERO KANBAN “CLUB DE LEONES QUITO”			
Lista de Tareas	Tareas en Proceso	Horas	Tareas Realizadas
Pruebas de funcionamiento al módulo de registro de usuarios (médico y paciente).	Creación de una instancia EC2 en AWS.	2	Realizar encuestas a médicos del área de Traumatología del “Club de Leones Quito”.
Pruebas de funcionamiento del módulo de historiales clínicos.	Instalación y codificación de WampServer.	3	Identificación de requerimientos funcionales (RF).
Pruebas de funcionamiento del módulo de registro y visualización de sugerencias de visitas médicas.	Implementación de la Capa de Datos.	15	Identificación de requerimientos no funcionales (RNF).
Pruebas de funcionamiento del módulo de registro de sugerencias por parte de los pacientes.	Implementación de la Capa de Negocio.	20	Selección del Proveedor de Cloud Computing
Pruebas de funcionamiento del módulo de carga de exámenes para el usuario paciente.	Codificación de las Clases.	20	Clasificación de los RF por módulos.
Pruebas de integración.	Codificación de los servicios Web.	20	Diseño de los diagramas de casos de uso.
Presentar resultados.	Implementación de la Capa de Presentación.	20	Diseño del diagrama relacional de base de datos.
			Diseño del diagrama de clases.
			Diseño de los servicios Web.
			Diseño de la capa de presentación.
			Diseño de las vistas de la aplicación móvil.

Figura 2.23. Tablero de Actividades Kanban de la Implementación.

2.2.2 Instalación de Herramientas Necesarias

En este apartado se presenta la instalación de todas las herramientas que permiten el desarrollo del prototipo; se mostrará los detalles más importantes de la instalación y de su respectiva configuración.

En la Figura 2.24 se puede observar cómo va a estar estructurado el servidor Windows Server en AWS y las herramientas necesarias que se deben instalar. Se utilizará el software denominado *WampServer*[33] que como se explicó en el **Capítulo 1 Apartado 1.3.4** dispone de algunos paquetes, entre ellos un sistema de gestión de base de datos MySQL y para la administración *phpMyAdmin*, la que almacenará la información concerniente al Sistema de Gestión Médica. Este software permite que también se almacenen los diferentes servicios web que se utilizará para permitir funcionalidades a la aplicación.

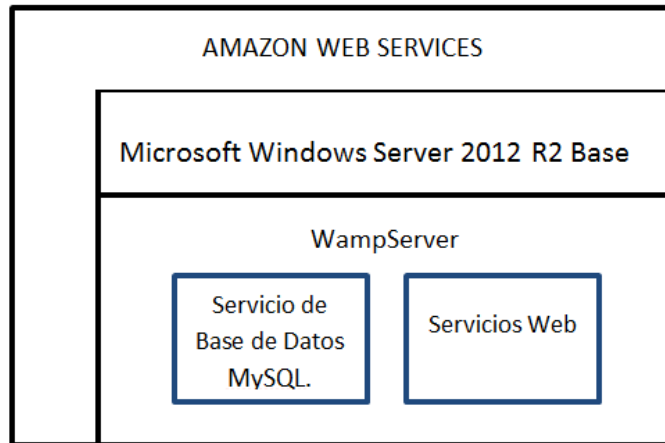


Figura 2.24. Arquitectura del servidor.

a. Creación de una Instancia EC2 en AWS

Como primer punto, se procede a crear una instancia de máquina virtual en la plataforma AWS. Para esto, en la página que presenta AWS [45] se escoge la opción “*Compute*” y se selecciona EC2⁹ debido a que esta opción es la que más se acopla al sistema permitiendo controlar y administrar las funciones del servidor, el sistema operativo y el software de implementación adicional [46]. La Figura 2.25 muestra la opción seleccionada.

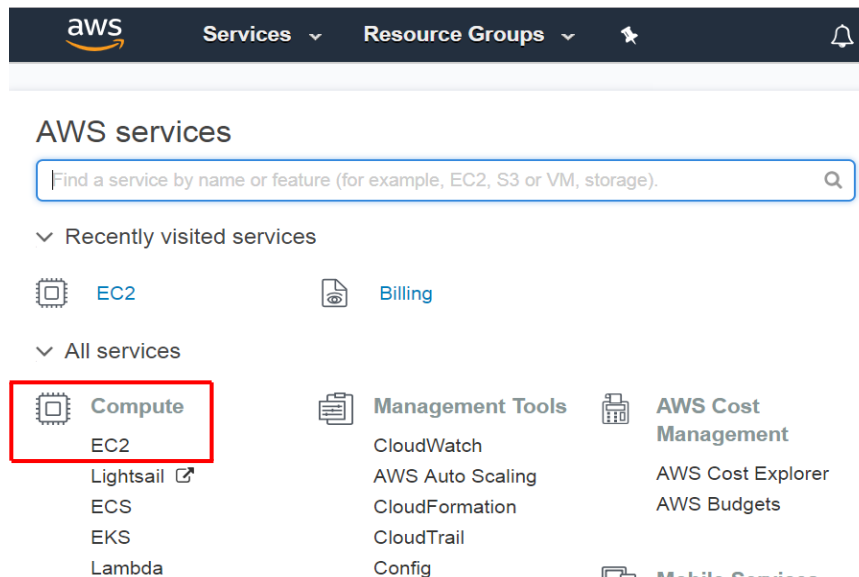


Figura 2.25. Selección del servicio para la instalación de una nueva máquina virtual.

Dentro de la consola de administración de AWS se escoge “*ejecutar instancia*” como se observa en la Figura 2.26.

⁹ EC2.- *Elastic Compute Cloud* forma parte central de la plataforma de computación en la nube para brindar un servicio de tipo IAAS

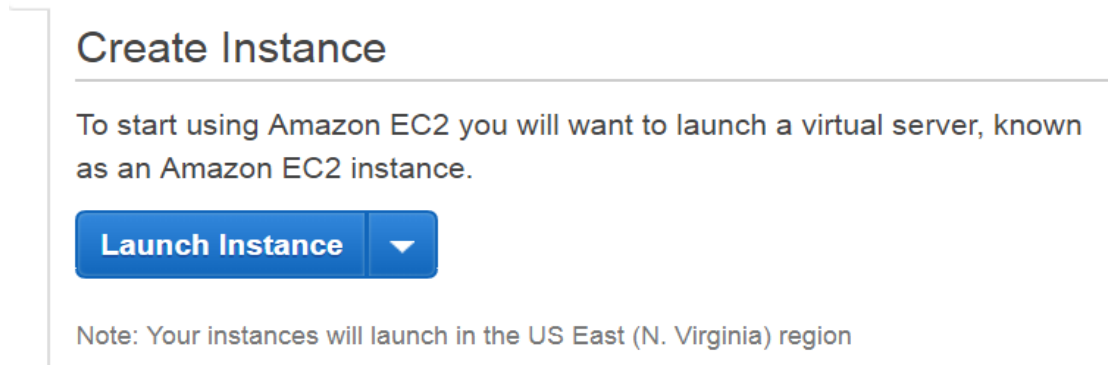


Figura 2.26. Botón para crear una instancia de AWS.

Con AWS, se puede especificar el software y las especificaciones de la instancia que se desea utilizar. En la Figura 2.27 se presentan varias opciones para elegir una imagen de MV de Amazon. Para el desarrollo de este Proyecto de Titulación se utilizará *Microsoft Windows Server 2012 R2 Base* que se resalta en el cuadro rojo.

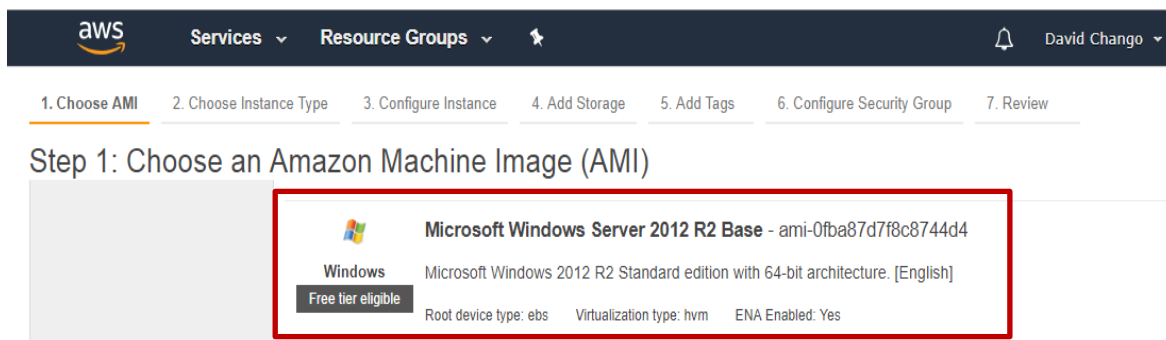


Figura 2.27. Opciones de Imágenes de AWS.

La Figura 2.28 muestra los tipos de instancias que dispone AWS donde, cada tipo de instancia dispone de diferentes tipos de configuraciones ya sea; CPU, memoria, almacenamiento y capacidad de red.

Step 2: Choose an Instance Type

Currently selected: t2.micro (Variable ECUs, 1 vCPUs, 2.5 GHz, Intel Xeon Family, 1 GiB memory, EBS only)

	Family	Type	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance	IPv6 Support
<input type="radio"/>	General purpose	t2.nano	1	0.5	EBS only	-	Low to Moderate	Yes
<input checked="" type="radio"/>	General purpose	t2.micro Free tier eligible	1	1	EBS only	-	Low to Moderate	Yes

Figura 2.28. Características del tipo de instancia escogida.

La Figura 2.29 muestra los detalles de la MV, el tipo de instancia (sistema operativo), el tamaño de almacenamiento y tamaño en memoria.

Step 7: Review Instance Launch

Please review your instance launch details. You can go back to edit changes for each section. Click **Launch** to assign a key pair to your instance and complete the launch process.

AMI Details [Edit AMI](#)

Microsoft Windows Server 2012 R2 Base - ami-2a9a1655
Microsoft Windows 2012 R2 Standard edition with 64-bit architecture. [English]
Root Device Type: ebs Virtualization type: hvm

Instance Type [Edit instance type](#)

Instance Type	ECUs	vCPUs	Memory (GiB)	Instance Storage (GB)	EBS-Optimized Available	Network Performance
t2.micro	Variable	1	1	EBS only	-	Low to Moderate

Storage [Edit storage](#)

Volume Type	Device	Snapshot	Size (GiB)	Volume Type	IOPS	Throughput (MB/s)	Delete on Termination	Encrypted
Root	/dev/sda1	snap-004c50f90619d0efe	30	gp2	100 / 3000	N/A	Yes	Not Encrypted

Figura 2.29. Resumen de las características del tipo de Servidor escogido.

Una vez comprobado los parámetros principales, se realiza la conexión al servidor a través del botón “*Conectar*” que se muestra en la Figura 2.30.

aws Servicios Grupos de recursos David Chango N. Virginia Apoyo

EC2 Dashboard Lanzar instancia Conectar Comportamiento

Filter by tags and attributes or search by keyword

Nombre	ID de instancia	Tipo de instanc	Zona de disponib	Estado de insta	Verificaciones d	Estado de alarm	DNS público (IPv4)	IP p
	i-0a8a879ed3f1bc4db	t2.micro	us-este-1d	corriendo	2/2 verifica...	Ninguna	ec2-52-90-86-156.com...	52.90

Instancia: i-0a8a879ed3f1bc4db DNS público: ec2-52-90-86-156.compute-1.amazonaws.com

Figura 2.30. Interfaz lista para conectar al servidor remoto en la nube AWS.

La Figura 2.31 muestra finalmente la interfaz del servidor en un ambiente *Cloud Computing* del proveedor AWS. La esquina superior de la Figura 2.31 detalla las respectivas direcciones IP's a las cuales se tendrá que direccionar desde la aplicación Android para que pueda acceder a todos los servicios que éste va a proveer.

```
Hostname : WIN-606HLLLEOP5
Instance ID : i-0a8a879ed3f1bc4db
Public IP Address : 52.90.86.156
Private IP Address : 172.31.92.233
Availability Zone : us-east-1d
Instance Size : t2.micro
Architecture : AMD64
```

Windows Server 2012 R2

12:40 AM 5/18/2018

Figura 2.31. Interfaz del Servidor en la Nube.

b. WampServer

En la MV previamente instalada y configurada, se instalará *WampServer*. A continuación, se describe el proceso de instalación de esta herramienta. En primer lugar, se accede a la página oficial [33] en donde se encuentra el software que se debe descargar de acuerdo al sistema operativo del servidor. La Figura 2.32 muestra que el *WampServer* que se utilizará es de 64 bits.

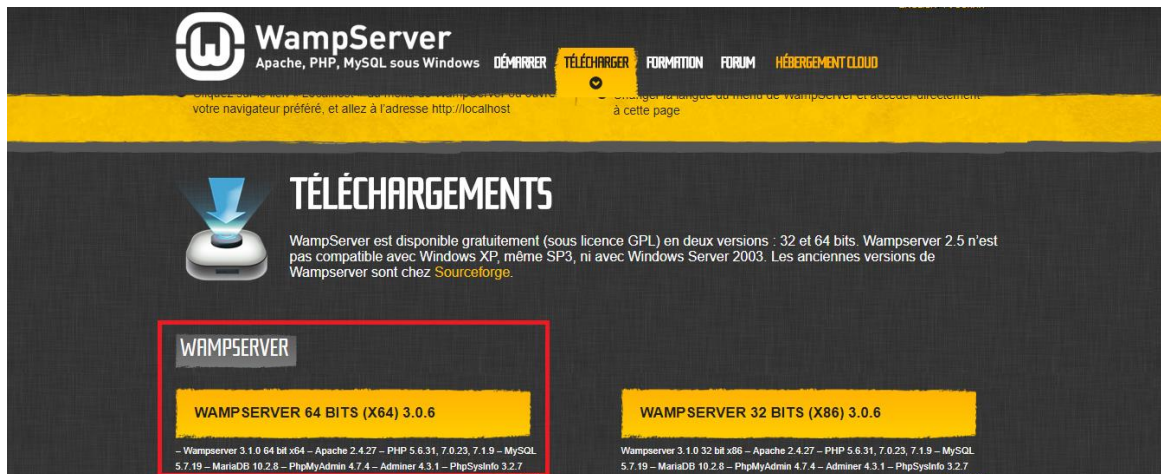


Figura 2.32. Descarga de WampServer de 64 bits.

Una vez descargado se ejecuta el programa para que pueda ser instalado en el servidor. La Figura 2.33 muestra que una vez instalado el *WampServer* se activan todos los servicios que contiene este software.

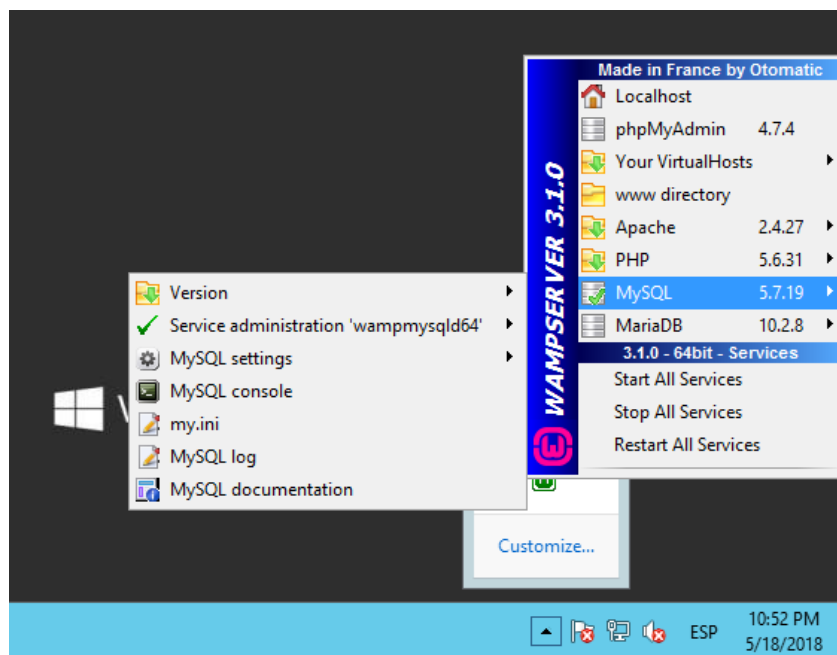


Figura 2.33. Servicios activos de WampServer.

c. NetBeans

A continuación, se describirá el proceso de instalación de esta herramienta. En primer lugar se accede a la página oficial [47] en donde se encuentra el software que se utiliza para el desarrollo de los servicios web en lenguaje PHP. Una vez descargado el paquete, simplemente se ejecuta el mismo y se visualiza la pantalla que se muestra en Figura 2.34.

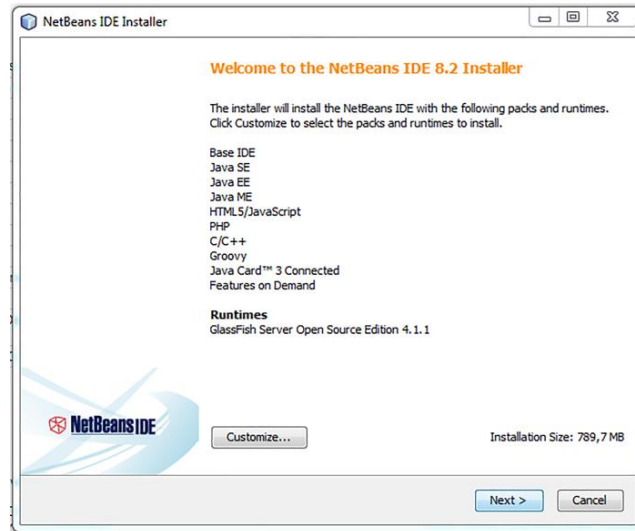


Figura 2.34. Instalación de NetBeans.

Una vez instalado el programa el paquete que interesa es el desarrollo de aplicaciones PHP como se muestra en la Figura 2.35.

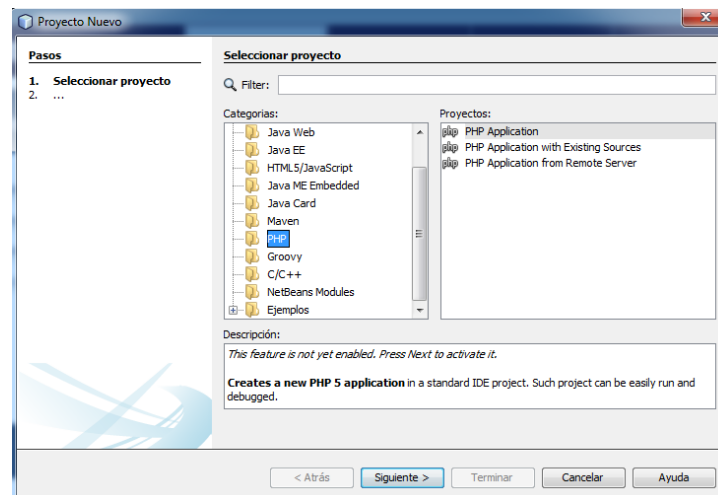


Figura 2.35. Aplicaciones de NetBeans.

Finalmente, en la Figura 2.36 se muestra la creación de la carpeta *WebServicesCloud* donde estarán almacenados cada uno de los servicios Web.

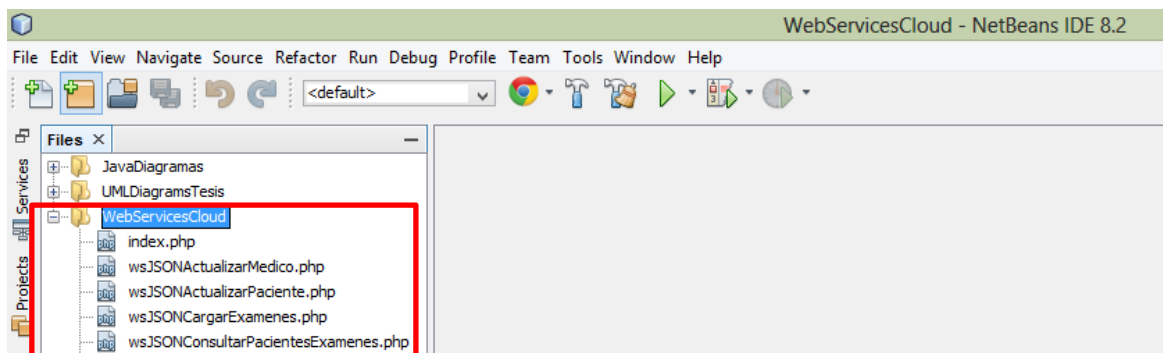


Figura 2.36. Creación del directorio para los servicios Web.

2.2.3 Codificación de la Capa de Base de Datos

A continuación, se presenta el detalle de la implementación de la capa de Base de Datos compuesta de diferentes tablas como se mencionó en el **Apartado 2.1.4** en la etapa de diseño. Para la codificación de las tablas se mostrará solamente un ejemplo de código.

Como se mencionó en el **Apartado 2.2.2**, WampServer dispone de la base de datos MySQL y su administrador *phpMyAdmin*; esta es una de las ventajas ya que no se tendrá que instalar por separado. A través de este software se realizará la creación de la base y de las tablas relacionadas como se indicó en la etapa de diseño.

Se procede a la codificación de la base de datos según el diseño realizado en el **Apartado 2.1**. El Código 2.1 corresponde a un fragmento del código ejecutado para la creación de la base de datos siguiendo el modelo relacional. Este código permite la creación de la tabla *Medico* misma que contiene los siguientes campos.

- *idmedico*: es el identificador que tendrá un usuario médico, este debe ser único para cada médico.
- *nombre*: nombre del médico que se ha registrado en la aplicación móvil.
- *username*: primera credencial que le permitirá el acceso a la aplicación móvil.
- *correo*: correo electrónico del usuario que se ha registrado en la aplicación móvil.
- *contrasenia*: segunda credencial necesaria que le permitirá el acceso a la aplicación móvil.
- *especialidad*: nombre de la especialidad en la que se encuentra laborando en la clínica.
- *telefono*: teléfono del usuario que se ha registrado en la aplicación móvil.

- *numeroConsultorio*: el número de consultorio que tiene el médico en el edificio donde se encuentra el “Club de Leones Quito”.
- *numeroLicencia*: número de referencia que autorice el ejercicio de la medicina.

En la línea 1 del Código 2.1 se crea la tabla en la base de datos *GestionMedica*, de la línea 2 a la 10 se definen el nombre de los campos, el tipo de dato, la longitud, además todos los campos tienen la propiedad *NOT NULL*, lo cual indica que no se aceptan valores nulos. Se observa que en la línea 11 se establece la clave primaria, que para esta tabla es el *idmedico*.

```

1 CREATE TABLE IF NOT EXISTS `GestionMedica`.`Medico` (
2   `idmedico` int(11) NOT NULL AUTO_INCREMENT,
3   `nombre` varchar(50) NOT NULL,
4   `username` varchar(50) NOT NULL,
5   `correo` varchar(50) NOT NULL,
6   `contrasenia` varchar(50) NOT NULL,
7   `especialidad` varchar(50) NOT NULL,
8   `telefono` int(50) NOT NULL,
9   `numeroConsultorio` varchar(50) NOT NULL,
10  `numeroLicencia` varchar(50) NOT NULL,
11  PRIMARY KEY (`idmedico`)
12 )

```

Código 2.1. Codificación de la tabla *Medico* en lenguaje SQL

El Código 2.2 muestra un ejemplo de consulta a la base de datos, la cual indica la consulta de la existencia de un médico en la tabla *Medico*. El *script* completo se presenta en el **Anexo III**.

```

1 SELECT * FROM MEDICO WHERE USERNAME="medico" and
  CONTRASENIA="medicoprueba"

```

Código 2.2. Ejemplo de consulta en SQL.

2.2.4 Codificación de la Capa de Negocio

A continuación, se presenta el detalle de la implementación de la capa de negocio compuesta de dos partes: la codificación de las clases y la codificación de los servicios web. Para la codificación de las clases se mostrará solamente un ejemplo de código, ya que el *script* completo se presenta en el **Anexo IV**.

a. Codificación de las Clases

Se procede a la codificación de clases siguiendo el diseño de clases presentado en el **Apartado 2.1.5**. Todas las clases descritas en el Diagrama de Clases serán codificadas en formato PHP y las mismas se encontrarán en el entorno *Cloud Computing*. En el Código 2.3 se presenta la implementación de la clase *Persona* con sus respectivos atributos; de la misma se heredan dos clases más: *Medico* y *Paciente*, las cuales tienen

los mismos campos de las tablas *Medico* y *Paciente* de la base de datos. En la línea 2 del Código 2.3 se muestra el nombre la clase. De la línea 4 a la 9 se muestra el nombre y el tipo de acceso de los atributos que para las clases son privados. Desde la línea 10 a la 24 muestran la forma de implementar un método *set()* y en la línea 34 a la 44 se muestra la forma de implementar el método *get()* en php mediante *function*¹⁰ que se usará en la clase para mostrar o modificar el valor de los atributos privados.

```

1  <?php
2  class Persona
3  {
4      private $id;
5      private $username;
6      private $contrasenia;
7      private $nombre;
8      private $correo;
9      private $telefono;
10     function setUsername($user) {
11         $this->username = $user;
12     }
13
14     function setPassword($pass) {
15         $this->contrasenia = $pass;
16     }
17
18     function setId($id) {
19         $this->id= $id;
20     }
21
22     function setNombre($nombre) {
23         $this->nombre = $nombre;
24     }
25
26
27
28
29
30
31
32
33
34     function getUsername() {
35         return $this->username;
36     }
37
38     function getPassword() {
39         return $this->contrasenia;
40     }
41
42     function getId() {
43         return $this->id;
44     }

```

Código 2.3. Código de Clase *Persona*.

En el Código 2.4, línea 2, se puede observar que la clase *Medico* es una clase heredada de la clase *Persona*. De la misma manera de la línea 4 a la 7 se muestra los atributos respectivos para esta clase. Igualmente, en esta clase se tendrá métodos *get()* y *set()* para acceder a los atributos de esta clase.

¹⁰ Permite la creación de funciones o métodos en php

```

1  <?php
2
3      class Medico extends Persona
4      {
5          private $idMedico;
6          private $especialidad;
7          private $numeroConsultorio;
8          private $numeroLicencia;
9
10         function setIdMedico($id){
11             $this->idMedico = $id;
12         }
13
14         function setEspecialidad($especialidad){
15             $this->especialidad= $especialidad;
16         }
17
18
19
20
21
22
23
24
25         function getIdMedico(){
26             return $this->idMedico;
27         }
28
29         function getEspecialidad(){
30             return $this->especialidad;
31         }

```

Código 2.4. Código de Clase *Medico*.

b. Codificación de los Servicios Web

Para la codificación de los servicios web se utilizará la herramienta NetBeans [32] ya que permite la creación de archivos PHP.

Cada vez que se necesite utilizar un servicio web y hacer consultas a la base de datos primero se hará la respectiva conexión con la base, para lo cual se tiene un archivo llamado *ConexionDB.php*; y al tener este archivo por separado solamente se lo llamará cada vez que se necesite la conexión a la base de datos. En el Código 2.5 de la línea 2 a la línea 8 se muestran los datos de conexión hacia la base de datos; la dirección IP donde se encuentra MySQL que en este caso, como se encuentra en el servidor será una dirección local del mismo, el nombre de la base de datos, el nombre de usuario y la contraseña de acceso.

```

1  <?php
2      $hostname_localhost="172.31.92.233";
3      $database_localhost="gestionmedica";
4      $username_localhost="root";
5      $password_localhost="";
6      $conexion=mysqli_connect($hostname_localhost,$username_localhost,
7      $password_localhost,$database_localhost) or
8      die ("ERROR AL CONECTARSE A LA BASE DE DATOS");
9  ?>

```

Código 2.5. Código de conexión para la base de datos.

En el Código 2.6 se muestra el *script* de la programación en PHP del servicio web *wsJSONLogin.php*; los datos que necesita el servicio web para realizar la consulta hacia la base de datos serán enviados desde la aplicación móvil.

En la línea 2 se llama al archivo *ConexionDB.php* para la conexión con la base de datos, en la línea 3 se crea un objeto JSON de tipo Array que será el encargado de almacenar la información que retorne el servicio web hacia la aplicación, los mismos que serán almacenados en una variable. En la línea 4 y 5 se guarda los datos de usuario y el contraseña. En la línea 6 se verifica que los datos guardados en las variables no estén vacías, si no es así se hará la consulta primero a la tabla *Medico* como indica la línea 7, con esto se verificará si los datos enviados corresponde a un usuario médico.

En la línea 8 se guardará el resultado de la consulta y de la conexión con la base. En el caso de que si corresponda a un médico se guarda los datos que se necesita en el objeto JSON como se muestra en las líneas 10 a la 12. Se realizará el mismo proceso en el caso de que sea un paciente como se indica en las líneas 15 a la 20. Además, en la línea 23, en el caso de no encontrar datos que correspondan a los enviados desde la aplicación se retorna un mensaje de USUARIO O CONTRASEÑA INCORRECTA. Finalmente, en la línea 27 se cierra la conexión con la base de datos.

```
<?PHP
2      require_once 'ConexionDB.php';
3      $json=array();
4      $contrasenia = $this->contrasenia;
5      $username = $this->username;
6      if($contrasenia != "" && $username != ""){
7          $consulta="select * from medico where username='{ $username}' and contrasenia='{ $contrasenia}'";
8          $resultado=mysqli_query($conexion,$consulta);
9          if($registro= mysqli_fetch_array($resultado)){
10             $json[0] = true;
11             $json[1] = $registro["idmedico"];
12             $json[2] = 'medico';
13         }
14     }else{
15         $consulta="select * from paciente where username= '{ $username}' and contrasenia='{ $contrasenia}'";
16         $resultado=mysqli_query($conexion,$consulta);
17         if($registro= mysqli_fetch_array($resultado)){
18             $json[0] = true;
19             $json[1] = $registro["idpaciente"];
20             $json[2] = 'paciente';
21         }else{
22             $json[0] = false;
23             $json[1] = "USUARIO O CONTRASEÑA INCORRECTA";
24             $json[2] = 'ninguno';
25         }
26     }
27     mysqli_close($conexion);
```

Código 2.6. Código de la implementación de un Servicio Web.

El *script* de todos los servicios web se muestra en el **Anexo V**.

2.2.5 Codificación de la Capa de Presentación

Para la codificación de la capa de Presentación se utilizará el software *Material Design*[48] de Android Studio que contiene una guía completa para el diseño visual, de movimiento y de interacción en plataformas y dispositivos.

A continuación, se presenta el detalle de la implementación de la capa de Presentación siguiendo el diseño realizado en el **Apartado 2.1.6**. Android Studio crea por defecto carpetas necesarias para el proyecto, una de ellas llamada *layout* donde se almacenará las interfaces graficas del mismo. En la Figura 2.37, se puede visualizar algunas de las interfaces que contiene todo el sistema. Para la codificación de la interfaz gráfica, se mostrará solamente un ejemplo de código, ya que el *script* completo estará en el **Anexo VI**. Además, en este mismo anexo se encontrará también el código completo desarrollado en Android Studio.

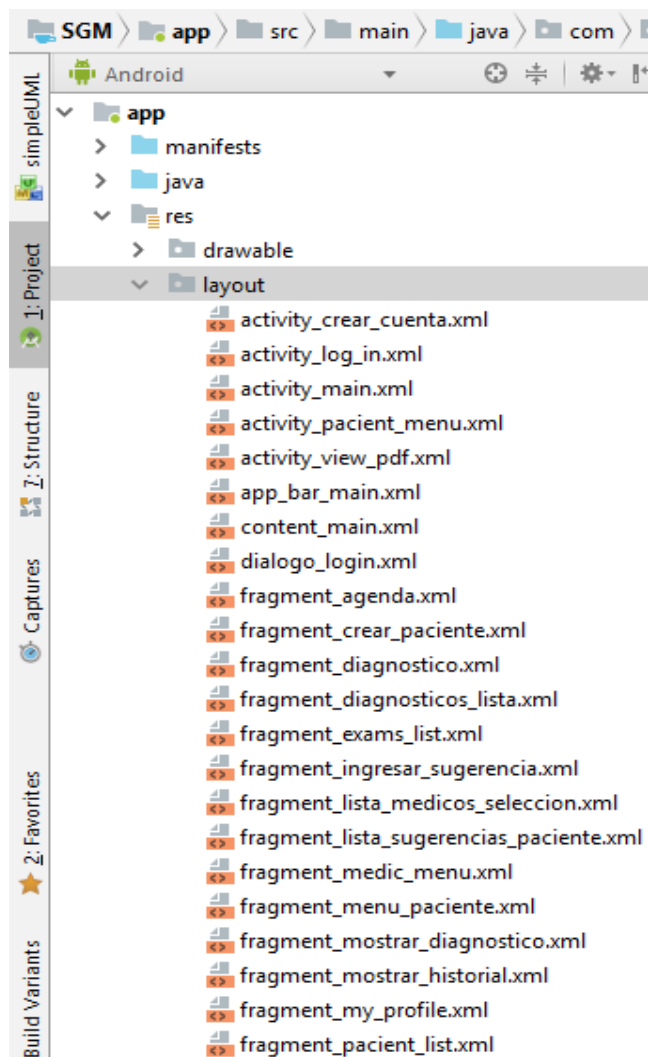


Figura 2.37. Interfaces del sistema de Gestión Médica en Android.

a. Codificación de las Vistas de la Aplicación Médico

En la Figura 2.38 se muestra las interfaces gráficas que visualizará el médico para que pueda registrarse en el Sistema de Gestión Médica. Para que un médico se pueda registrar, el sistema debe validar que efectivamente es un usuario médico, para lo cual este dispondrá de un usuario y contraseña que servirá únicamente para el registro.

Figure 2.38 shows the medical user registration interface. It includes a login section with 'Username' and 'Password' fields and 'INGRESAR' and 'CREAR CUENTA' buttons. The central registration section is titled 'Acceso Registro Médicos' and contains a text input field with 'medico' and a password field with masked characters '.....', accompanied by a red 'OK' button. The right side features a registration form with fields for 'Nombres', 'Username', 'Correo Electrónico', 'Contraseña', 'Especialidad', 'Teléfono', 'Número de Consultorio', and 'Número de Licencia', with a blue 'REGISTRARSE' button at the bottom.

Figura 2.38. Pantalla de registro usuario Médico.

El Código 2.7 muestra una porción de código para la implementación de las diferentes interfaces gráficas. Android maneja la creación de *layouts* en modo texto o mediante código XML¹¹.

El Código 2.7 muestra algunos elementos creados, así se tiene: en la línea 1 la declaración de un *LinearLayout* que es un elemento que permite colocar a los componentes gráficos uno detrás del otro; en la línea 4 y 5 se define el tamaño de este componente; en la línea 6 se define la orientación en la que los elementos serán ubicados que en este caso serán puestos en posición vertical; en la línea 8 se indica el nombre de la pantalla; en la línea 11 se define un *ScrollView* el cual es un elemento que permite el desplazamiento de los componentes gráficos que no alcancen en la pantalla principal y de esta manera facilitar la navegación.

En la línea 15 se muestra la creación de un nuevo *LinearLayout*; en la línea 19 se define un *TextInputLayout*, que es un elemento que muestra una etiqueta flotante cuando el usuario ingresa texto; en las líneas 19 a la 30 se definen datos como: alto, ancho, separación entre componentes, etc.

¹¹ XML es un lenguaje que permite la organización y el etiquetado de documentos.

```

1 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:app="http://schemas.android.com/apk/res-auto"
3   xmlns:tools="http://schemas.android.com/tools"
4   android:layout_width="match_parent"
5   android:layout_height="match_parent"
6   android:orientation="vertical"
7   tools:context="com.example.david.sistemagestionmedica.CrearCuentaFragment"
8   android:id="@+id/CrearCuentaFragment">
9
10  <!-- TODO: Update blank fragment layout -->
11  <ScrollView
12    android:layout_width="match_parent"
13    android:layout_height="match_parent">
14
15    <LinearLayout
16      android:layout_width="match_parent"
17      android:layout_height="match_parent"
18      android:orientation="vertical">
19      <android.support.design.widget.TextInputLayout
20        android:layout_width="332dp"
21        android:layout_height="51dp"
22        android:layout_marginBottom="8dp"
23        android:layout_marginEnd="8dp"
24        android:layout_marginStart="8dp"
25        android:layout_marginTop="8dp"
26        app:layout_constraintBottom_toBottomOf="parent"
27        app:layout_constraintEnd_toEndOf="parent"
28        app:layout_constraintStart_toStartOf="parent"
29        app:layout_constraintTop_toTopOf="parent"
30        app:layout_constraintVertical_bias="0.0">

```

Código 2.7. Código XML para la implementación gráfica.

La Figura 2.39 muestra las interfaces gráficas que visualizará el médico para registrar un paciente nuevo.

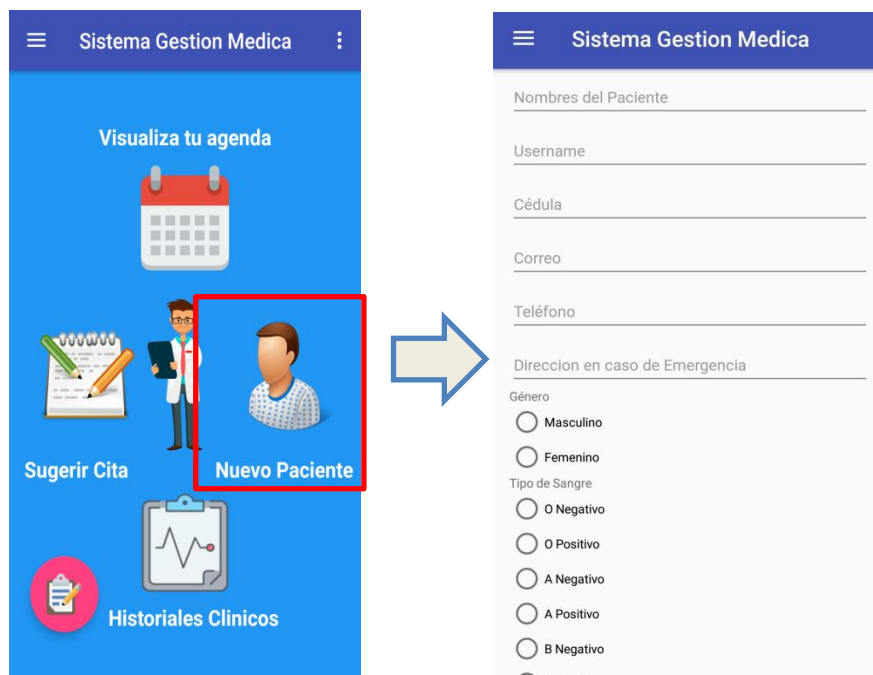


Figura 2.39. Pantalla de registro paciente nuevo.

La Figura 2.40 muestra las interfaces para visualizar los historiales clínicos de cada paciente, y el menú que permite mostrar exámenes, cargar exámenes, mostrar historial y eliminar paciente.

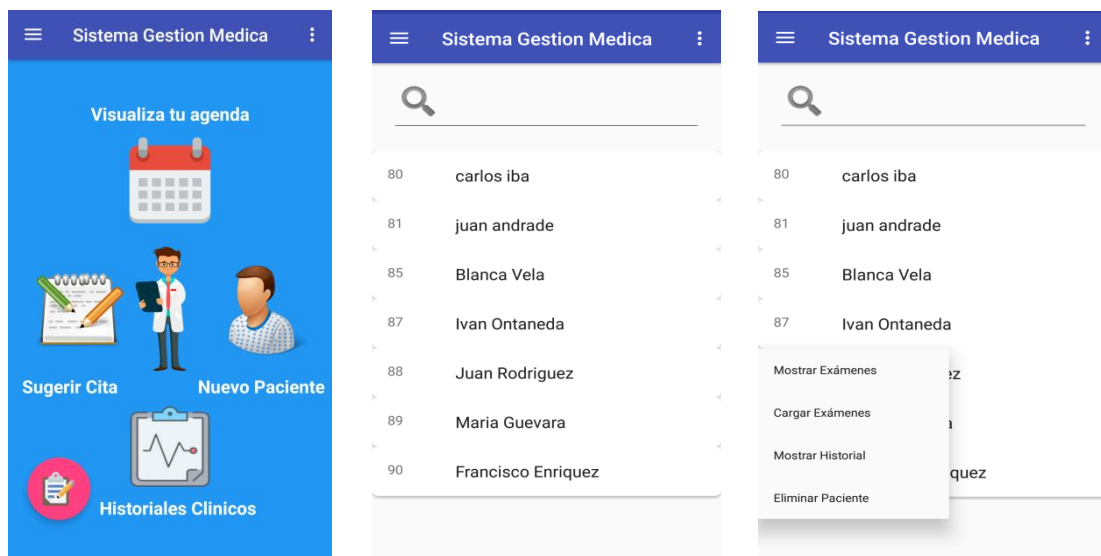


Figura 2.40. Pantallas para la Gestión de Historiales Clínicos.

La Figura 2.41 muestra las interfaces para las opciones de subir exámenes y mostrar los mismos de acuerdo al paciente seleccionado, además se muestra un *pop menu* con las opciones de descargar y eliminar.

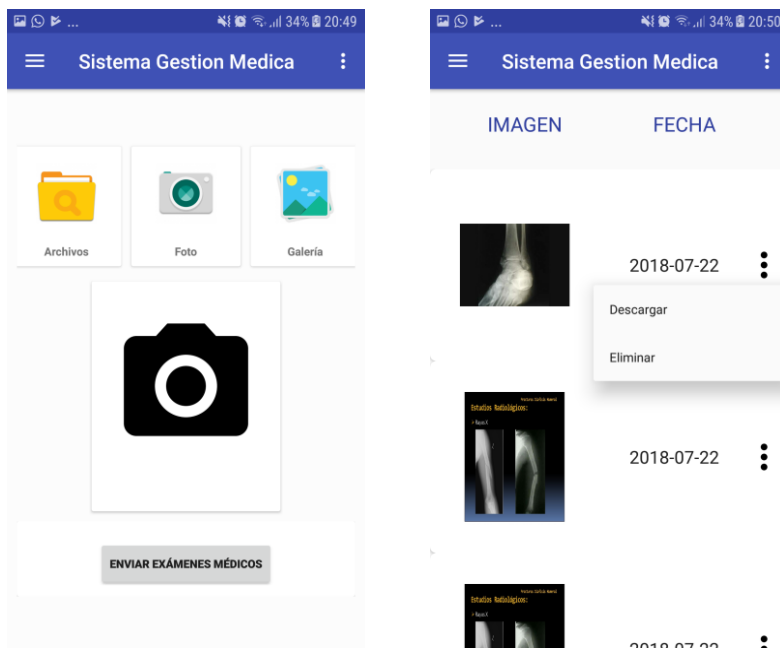


Figura 2.41. Pantallas de Cargar Exámenes y Mostrar Exámenes.

La Figura 2.42 indica las interfaces para la opción de mostrar el historial del paciente seleccionado, así como la creación de diagnósticos y la visualización del historial completo.

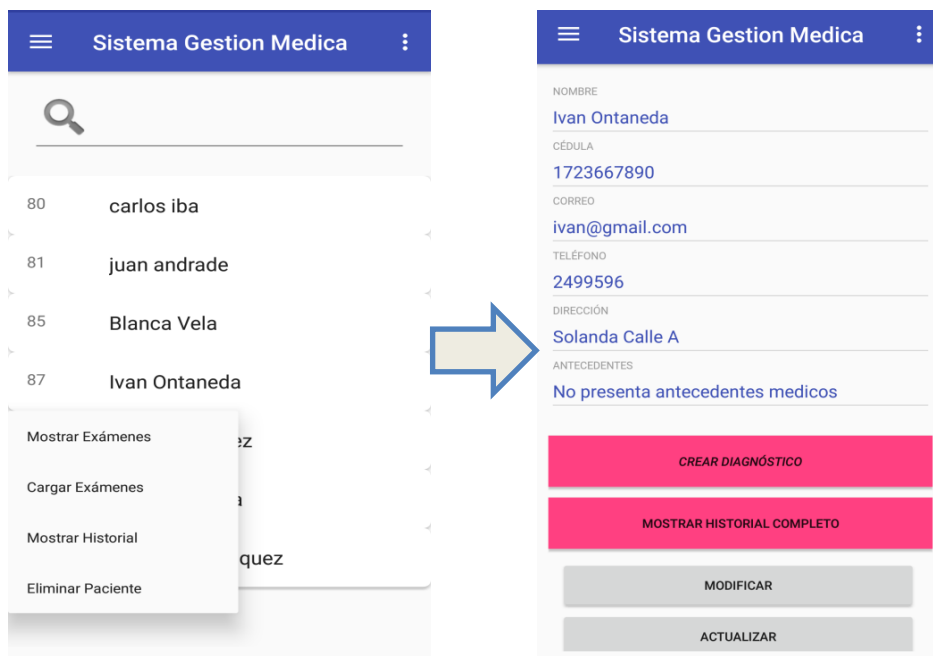


Figura 2.42. Pantalla de mostrar historial y creación de diagnósticos.

Dentro de esta interfaz se cuenta con las opciones de crear diagnóstico y mostrar el historial completo, como se observa en la Figura 2.43.



Figura 2.43. Pantallas de Crear Diagnóstico y Mostrar Historial Completo

La Figura 2.44 muestra la interfaz para la opción Eliminar Paciente.

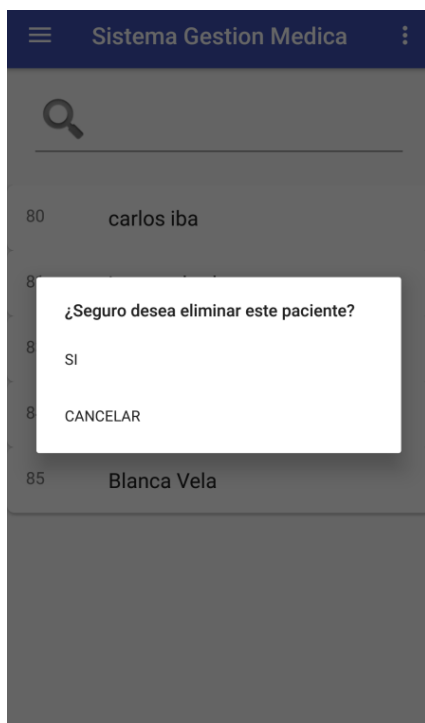


Figura 2.44. Pantalla Eliminar Paciente.

La Figura 2.45 muestra las interfaces respectivas para que el médico sugiera un día para la próxima visita.

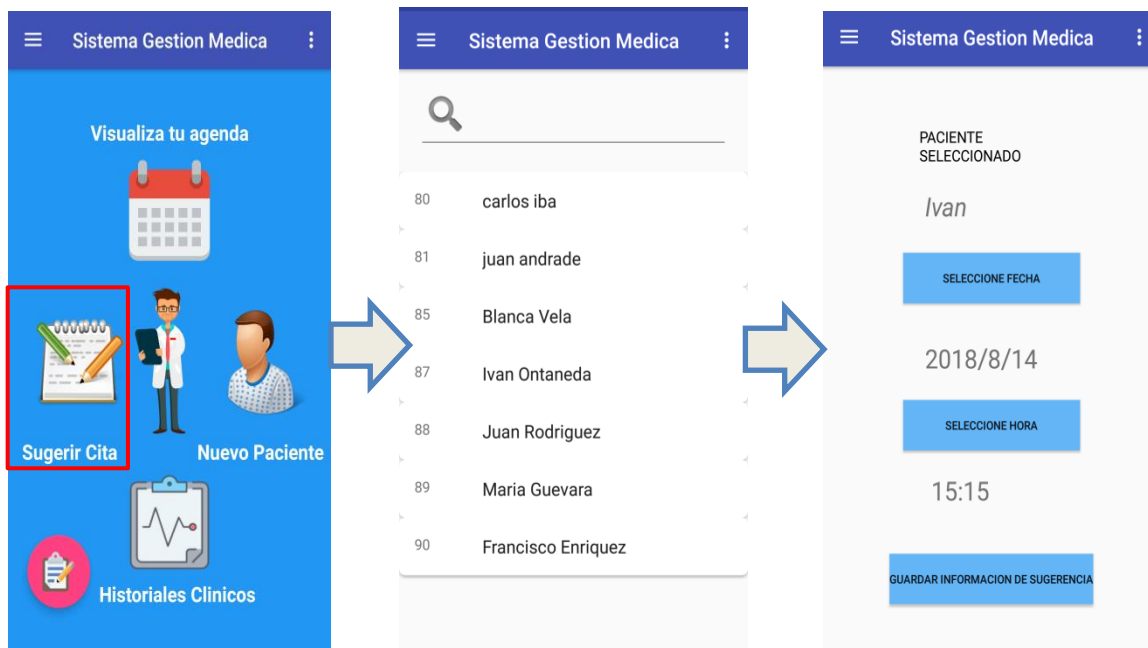


Figura 2.45. Pantallas para la sugerencia de citas médicas.

La Figura 2.46 muestra las interfaces respectivas para que el médico pueda visualizar las posibles visitas de sus pacientes.



Figura 2.46. Pantallas para la visualización de citas médicas.

La Figura 2.47 muestra un menú lateral donde el médico podrá visualizar sus datos personales y si desea puede modificar y actualizar la información.

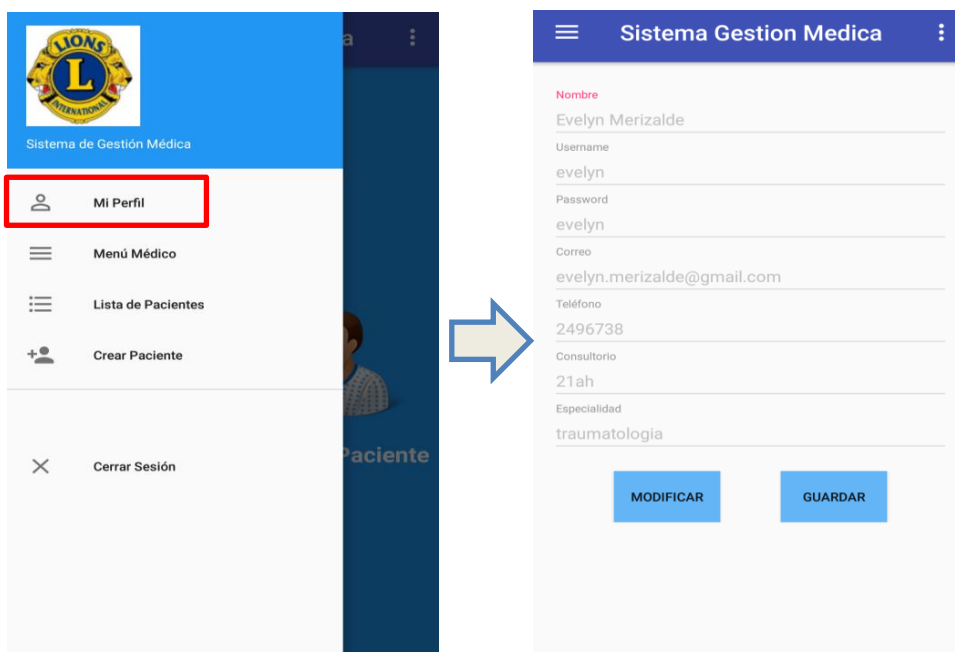


Figura 2.47. Pantalla de visualización Mi Perfil

b. Codificación de las Vistas de la Aplicación Paciente

La Figura 2.48 es la interfaz gráfica principal que dispondrá un usuario con el rol de paciente, además se muestra la interfaz donde visualizará su información personal. Se puede observar las diferentes actividades que podrá realizar como se definió en el **Apartado 2.1.2.**

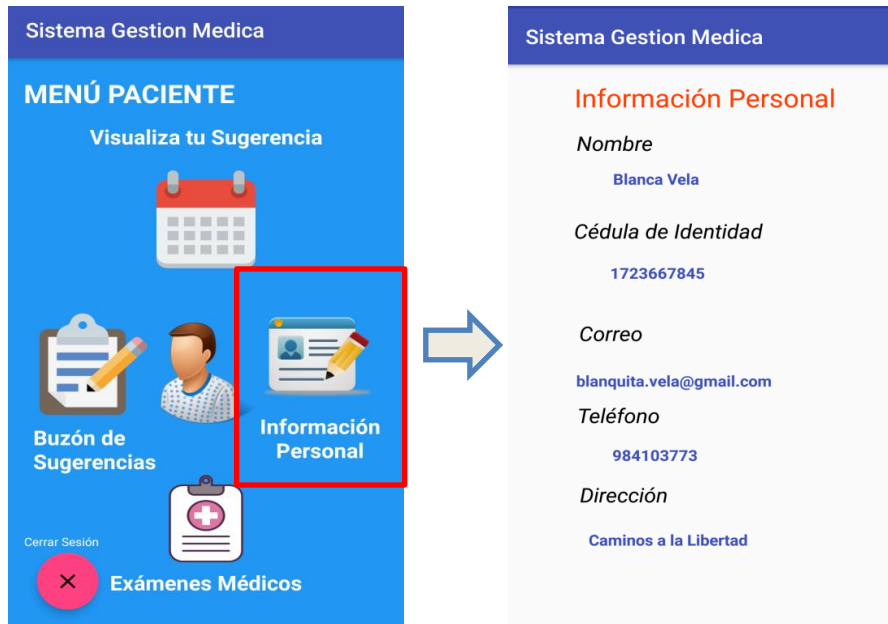


Figura 2.48. Pantalla Principal Usuario Paciente y visualización de información personal.

La Figura 2.49 indica la interfaz para la visualización de la sugerencia de cita médica.



Figura 2.49. Pantallas de visualización de sugerencia.

Para dejar sugerencias de la atención médica, comentarios o quejas se tiene las interfaces presentadas en la Figura 2.50.

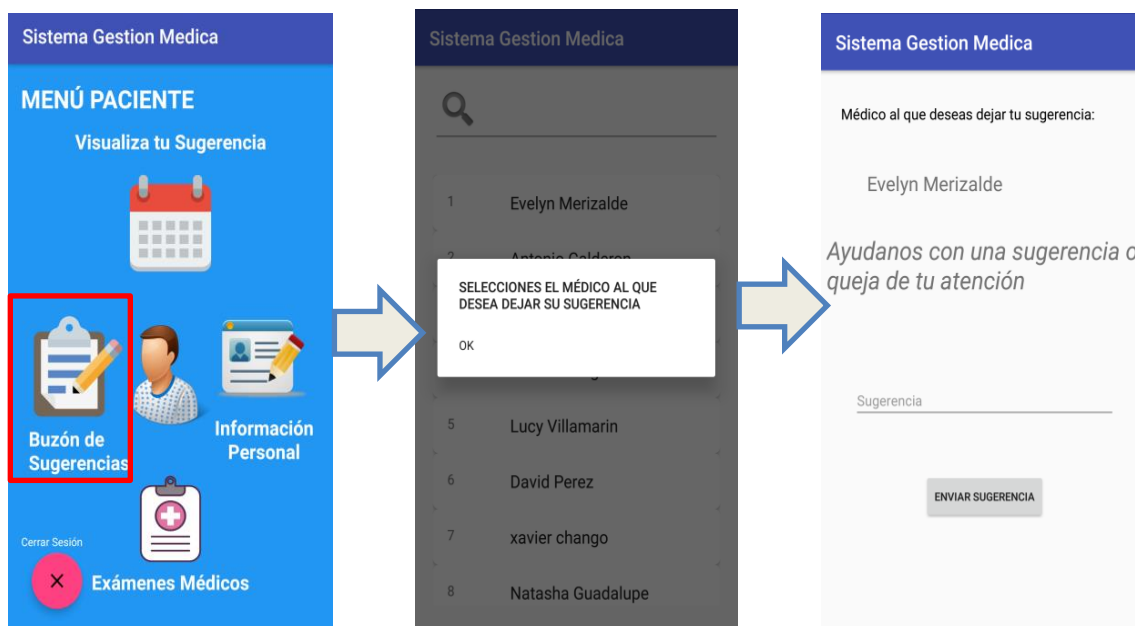


Figura 2.50. Pantallas de sugerencias o quejas.

La Figura 2.51 presenta las interfaces que permiten cargar los exámenes médicos. Las interfaces deben ser los más intuitivas posibles, ya que esto hace que el usuario final quiera utilizar la aplicación.

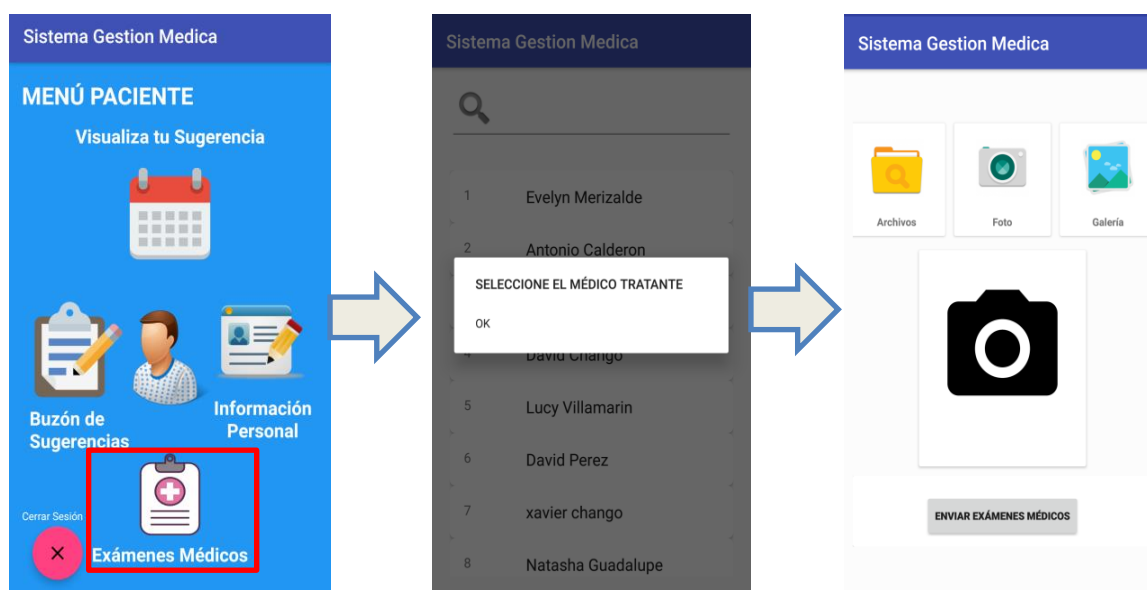


Figura 2.51. Pantallas para cargar Exámenes Médicos.

3. RESULTADOS Y DISCUSIÓN

El presente capítulo muestra las pruebas realizadas sobre el prototipo implementado. Se comenzará actualizando el respectivo Tablero de Kanban con las actividades para esta etapa final. Por último, se validará que los requerimientos funcionales, como los no funcionales se cumplan como se definió en el **Capítulo 2 Apartado 2.1.2**. Por último, se analizarán los resultados obtenidos.

3.1 Tablero de Actividades Kanban

En la Figura 3.1 se muestra el tablero Kanban actualizado para este capítulo. Las “*Tareas en Proceso*” son concernientes a las pruebas de funcionalidad de cada uno de los módulos; las “*Tareas Realizadas*” indican que la fase de diseño y de implementación se ha cumplido en su totalidad.

TABLERO KANBAN “CLUB DE LEONES QUITO”			
Lista de Tareas	Tareas en Proceso	Horas	Tareas Realizadas
	Pruebas de funcionamiento al módulo de registro de usuarios (médico y paciente).	5	Selección de herramientas a utilizarse
	Pruebas de funcionamiento del módulo de registro y visualización de sugerencias de visitas médicas.	5	Realizar encuestas a médicos del área de Traumatología del “Club de Leones Quito”
	Pruebas de funcionamiento del módulo de registro de sugerencias y quejas por parte de los pacientes.	5	Identificación de requerimientos funcionales (RF).
	Pruebas de funcionamiento del módulo de historiales clínicos.	5	Identificación de requerimientos no funcionales (RNF).
	Pruebas de funcionamiento del módulo de usuario Paciente	5	Clasificación de los RF por módulos.
	Pruebas de integración.	5	Diseño de los diagramas de casos de uso.
	Presentar resultados.	10	Diseño del diagrama relacional de base de datos.
			Diseño del diagrama de clases.
			Diseño del servicio Web.
			Diseño de la capa de presentación.
			Diseño de las vistas de la aplicación móvil.
			Creación de una instancia EC2 en AWS.
			Instalación y codificación de WampServer.
			Implementación de la Capa de Datos.
			Implementación de la Capa de Negocio.
			Codificación de las Clases.
			Codificación de los servicios Web.
			Implementación de la Capa de Presentación.

Figura 3.1. Tablero Kanban para la etapa de resultados y pruebas.

3.2 Pruebas de Validación de Requerimientos Funcionales

En este apartado se valida el cumplimiento de los Requerimientos Funcionales en cada uno de los subsistemas descritos en el **Capítulo 2 Apartado 2.1.5**.

3.2.1 Pruebas de Validación del Subsistema 01: Creación de Perfil

En este Subsistema se verificará el correcto funcionamiento del registro de médicos. La Tabla 3.1 muestra el RF para el registro de un usuario médico.

Tabla 3.1. Descripción del RFM01 por comprobar funcionamiento.

SUBSISTEMA	RF	CASO DE USO	DESCRIPCIÓN
SB01	RFM01	CU01	Registrar médicos.

La Figura 3.2 muestra un mensaje de validación para que todos los campos sean ingresados, con eso se garantiza que el usuario llene todos los datos necesarios.

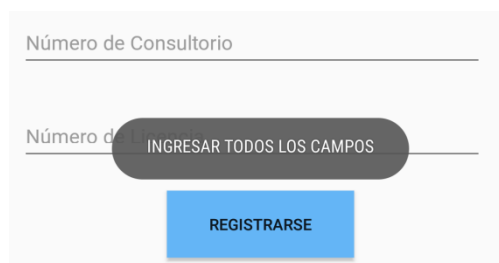


Figura 3.2. Mensaje de validación de campos.

Como se observa en la Figura 3.3 después de validar todos los campos ingresados se presenta un mensaje de confirmación que se ha registrado exitosamente. La Figura 3.3 muestra el cumplimiento del requerimiento funcional: RFM01



Figura 3.3. Mensaje de confirmación de registro de nuevo médico.

3.2.2 Pruebas de Validación del Subsistema 02: Gestión de Información Personal

En este Subsistema se verificará el correcto funcionamiento de la autenticación y gestión de la información personal del médico. La Tabla 3.2 muestra los RF para cumplir con el proceso de gestión de información personal.

Tabla 3.2. Descripción de los RF asociados al SB02 por comprobar funcionamiento.

SUBSISTEMA	RF	CASO DE USO	DESCRIPCIÓN
SB02	RFM02	CU02	Loguear médico
	RFM03	CU02	Visualizar información personal médico
	RFM04	CU02	Modificar información personal médico

Como se observa en la Figura 3.4 después de validar que el usuario efectivamente existe se da acceso al Sistema de Gestión Médica como usuario médico. En la Figura 3.4 se muestra el cumplimiento del requerimiento funcional: RFM02.

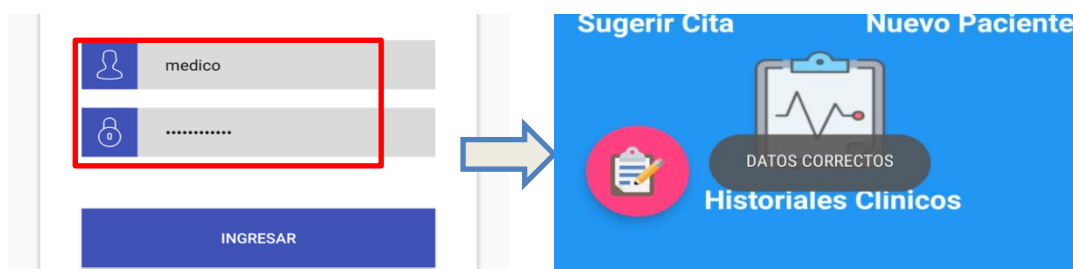


Figura 3.4. Acceso usuario médico.

La Figura 3.5 muestra un mensaje de error en el caso de que el médico haya ingresado en forma incorrecta el username o la contraseña.

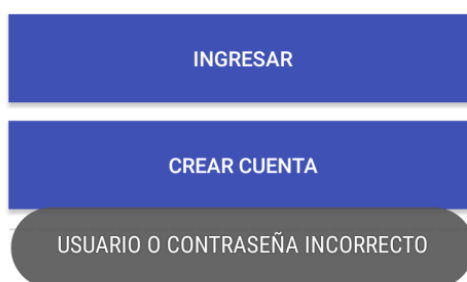


Figura 3.5. Mensaje de validación de datos de usuario.

Una vez que haya accedido el médico tiene la opción de visualizar su perfil. En la Figura 3.6 se muestra el cumplimiento de los requerimientos funcionales: RFM03 y RFM04, los cuales son la visualización y modificación de la información personal respectivamente.

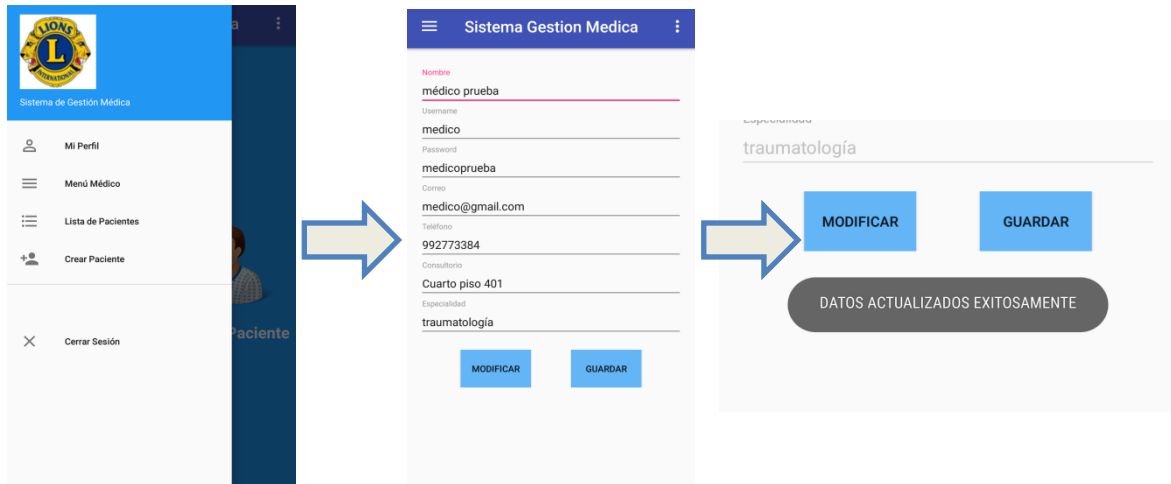


Figura 3.6. Visualización y modificación de datos de usuario médico.

Si el médico al momento de modificar su información se olvida de completar algún campo o lo deja vacío se mostrará un mensaje de error como se muestra en la Figura 3.7.



Figura 3.7. Mensaje de error al momento de actualizar.

3.2.3 Pruebas de Validación del Subsistema 03: Gestión de Pacientes

En este Subsistema se verificará el correcto funcionamiento de la gestión de pacientes. La Tabla 3.3 muestra los requerimientos que se debe cumplir y las actividades que debe realizar el médico.

Tabla 3.3. Descripción de los RF asociados al SB03 por comprobar funcionamiento.

SUBSISTEMA	RF ASOCIADO	CASO DE USO	DESCRIPCIÓN
SB03	RFM05	CU03	Agregar paciente
	RFM06	CU03	Modificar paciente
	RFM07	CU03	Eliminar paciente
	RFM08	CU03	Buscar paciente
	RFM09	CU03	Visualización de todos los pacientes

Una vez que el médico haya accedido al sistema una de las principales actividades es la gestión de pacientes: agregar, modificar y eliminar. En la Figura 3.8 se muestra el cumplimiento del requerimiento funcional: RFM05, el cual es agregar paciente.

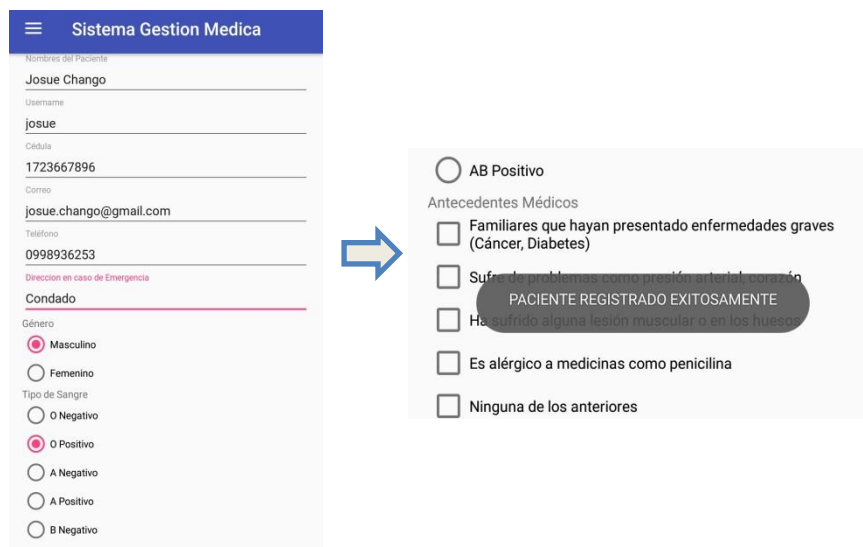


Figura 3.8. Mensaje de paciente registrado exitosamente.

En el caso de que el médico cree un paciente que ya está registrado el sistema debe mostrar un mensaje de que ya existe dicho paciente, así se evitará tener pacientes duplicados. La Figura 3.9 muestra el mensaje de advertencia de paciente ya creado.

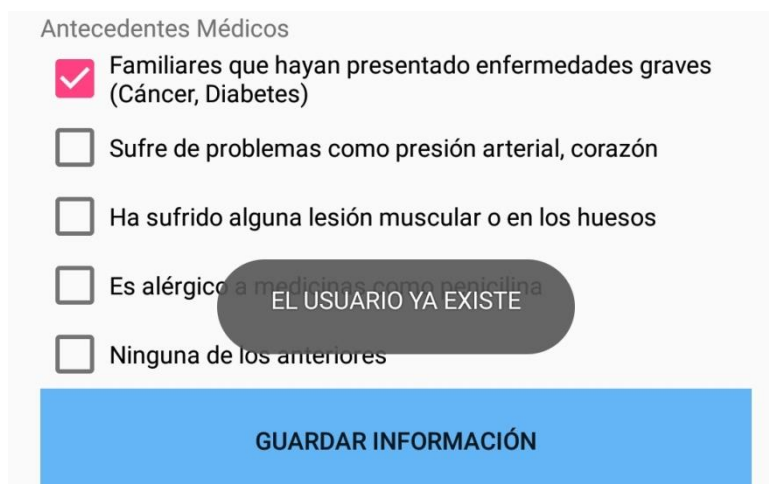


Figura 3.9. Mensaje de paciente ya creado.

Una vez que el médico ha registrado un paciente nuevo y en el caso de que se quiera modificar la información personal, el médico lo podrá hacer. La Figura 3.10 muestra la actualización de información del correo, teléfono, dirección y antecedentes del paciente ya creado y a la vez un mensaje de confirmación del cambio realizado.



Figura 3.10. Actualización información paciente.

En la Figura 3.11 se muestra el cumplimiento del requerimiento funcional: RFM07, el cual es eliminar paciente.

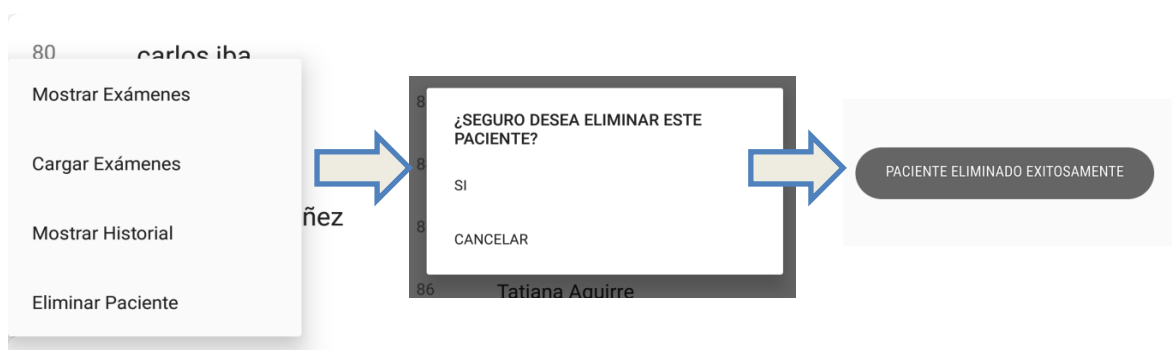


Figura 3.11. Prueba de eliminación de pacientes.

El médico puede visualizar toda la lista de pacientes y en la caso de existir muchos tenga la facilidad de buscarlos. La Figura 3.12 muestra el cumplimiento de los requerimientos funcionales: RFM08 y RFM09, los cuales son buscar un paciente y visualización de todos los pacientes respectivamente

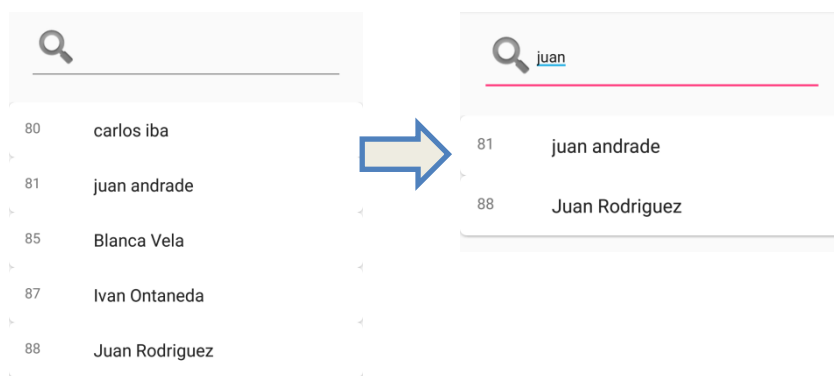


Figura 3.12. Buscador de pacientes.

3.2.4 Pruebas de Validación del Subsistema 04: Gestión de atención al paciente

En este Subsistema se verificará el correcto funcionamiento de la gestión de sugerencias de citas médicas. La Tabla 3.4 muestra los requerimientos que se deben cumplir.

Tabla 3.4. Descripción de los RF asociados al SB04 por comprobar funcionamiento.

SUBSISTEMA	RF ASOCIADO	CASO DE USO	DESCRIPCIÓN
SB04	RFM10	CU04	Agregar sugerencias de visita
	RFM11	CU04	Visualizar agenda
	RFM12	CU04	Visualizar sugerencias de la atención

En la Figura 3.13 se muestra el proceso de cómo el médico puede realizar una sugerencia de visita. Además, el médico puede visualizar todas las sugerencias que ha realizado. La Figura 3.13 muestra el cumplimiento de los casos de uso: RFM10 y RFM11, los cuales son agregar sugerencias de visita y visualización de agenda.

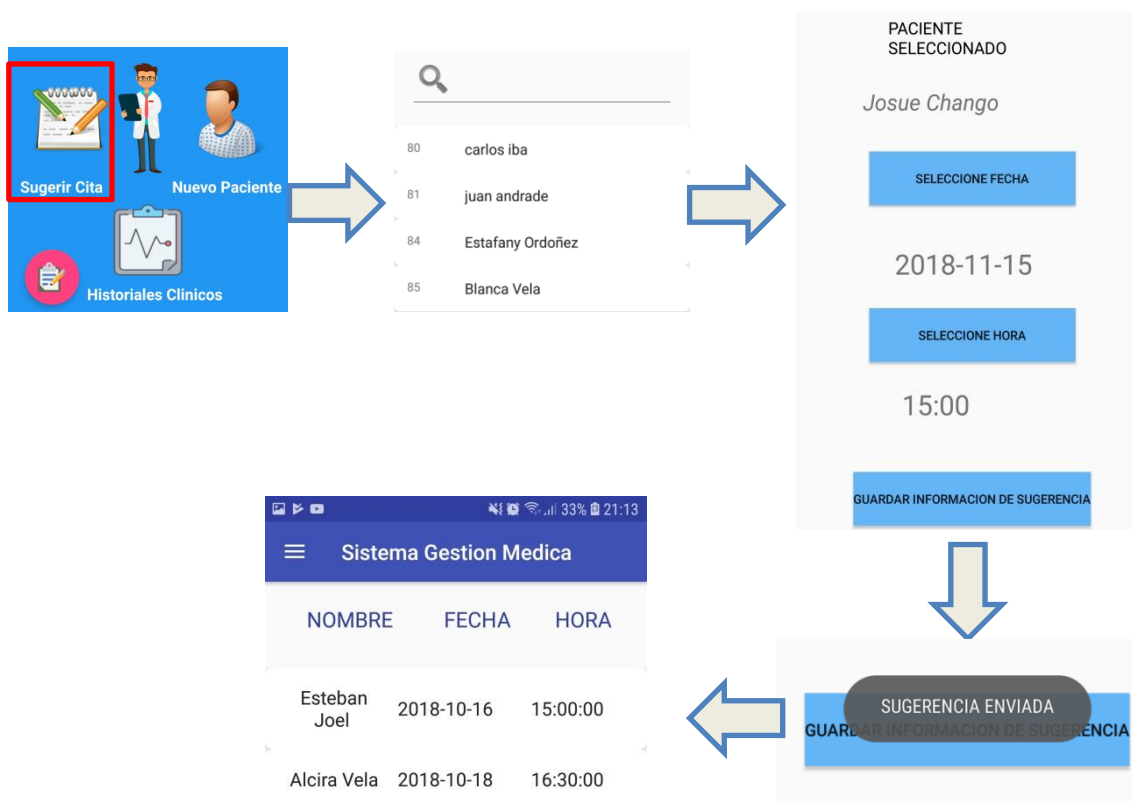


Figura 3.13. Sugerencia de visitas médicas y agenda.

En el caso de que el médico no haya sugerido ninguna cita médica se mostrará un mensaje como se indica en la Figura 3.14.

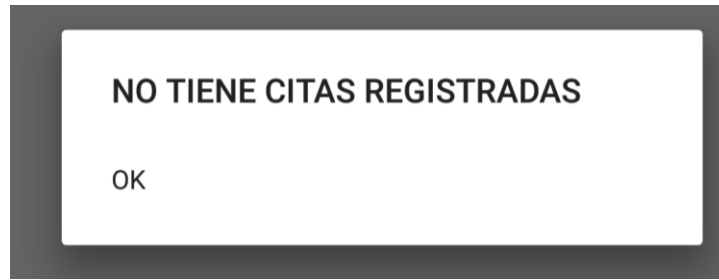


Figura 3.14. Mensaje de notificación de visitas médicas.

Los médicos puedan visualizar las sugerencias de sus pacientes. En la Figura 3.15 se muestra las sugerencias que han hecho los pacientes sobre el servicio médico. La Figura 3.15 muestra el cumplimiento del requerimiento funcional: RFM12, visualización de sugerencias o quejas.

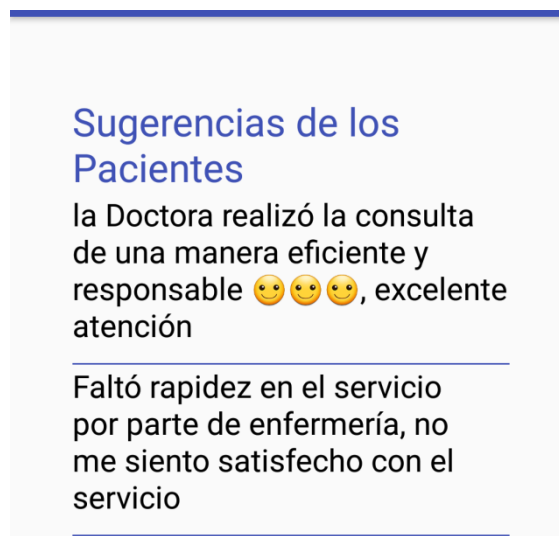


Figura 3.15. Sugerencias y quejas de pacientes.

Si los pacientes no han registrado ninguna sugerencia, se mostrará un mensaje de notificación, tal como se muestra en la Figura 3.16.



Figura 3.16. Mensaje de notificación de sugerencias.

3.2.5 Pruebas de Validación del Subsistema 05: Gestión de historiales clínicos

En este Subsistema se verificará el manejo de la información de los pacientes: carga de exámenes, creación de diagnósticos, mostrar historiales completos. La Tabla 3.5 muestra el caso de uso y los requerimientos funcionales que se debe cumplir.

Tabla 3.5. Descripción de los RF asociados al SB05 por comprobar funcionamiento.

SUBSISTEMA	RF ASOCIADO	CASO DE USO	DESCRIPCIÓN
SB05	RFM13	CU05	Cargar exámenes médicos.
	RFM14	CU05	Eliminar exámenes médicos.
	RFM15	CU05	Crear Historial.
	RFM16	CU05	Crear diagnóstico.
	RFM17	CU05	Solicitar exámenes médicos.
	RFM18	CU05	Añadir descripción (estado de salud).
	RFM19	CU05	Añadir receta.
	RFM20	CU05	Generar Reporte.

El médico está en la capacidad de cargar los exámenes médicos de sus pacientes y de la misma manera eliminarlos. La Figura 3.18 muestra el cumplimiento del requerimiento funcional RFM13 el cual es cargar exámenes médicos, éstos pueden ser archivos .pdf o fotos ya sean tomadas de la cámara o accediendo a la galería del dispositivo.

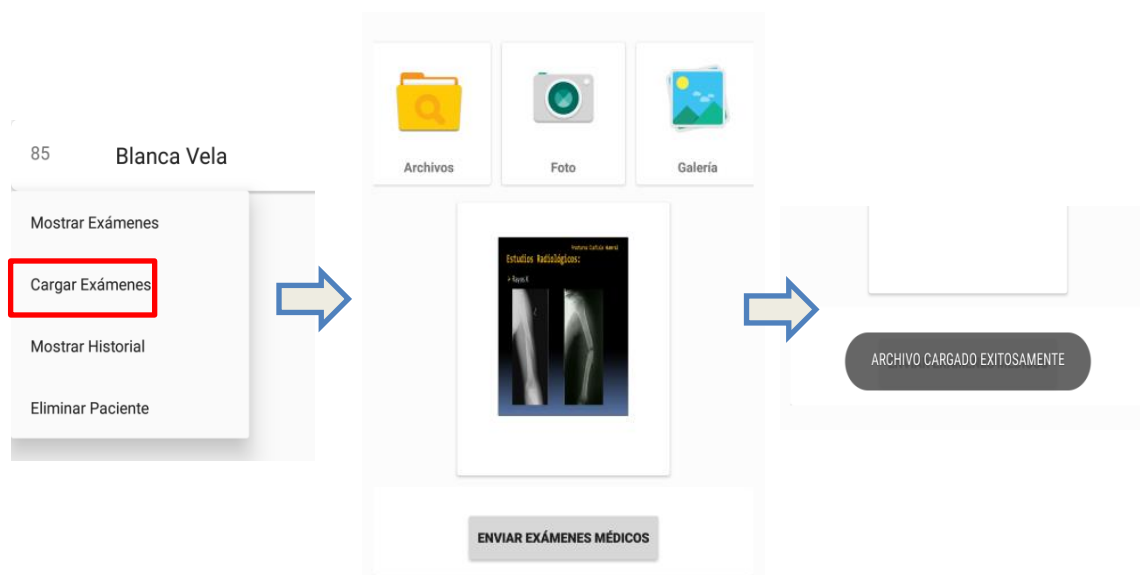


Figura 3.17. Carga de exámenes médicos.

La Figura 3.18 muestra el cumplimiento del requerimiento funcional RFM14 el cual es eliminar exámenes médicos.



Figura 3.18. Eliminación de exámenes médicos.

Al momento de registrar un nuevo paciente automáticamente se crea un historial para el mismo. En este historial se mostrará los diagnósticos previamente creados. La Figura 3.19 muestra el cumplimiento de los requerimientos funcionales: RFM15 y RFM16, los cuales son creación de historial y creación de diagnósticos.

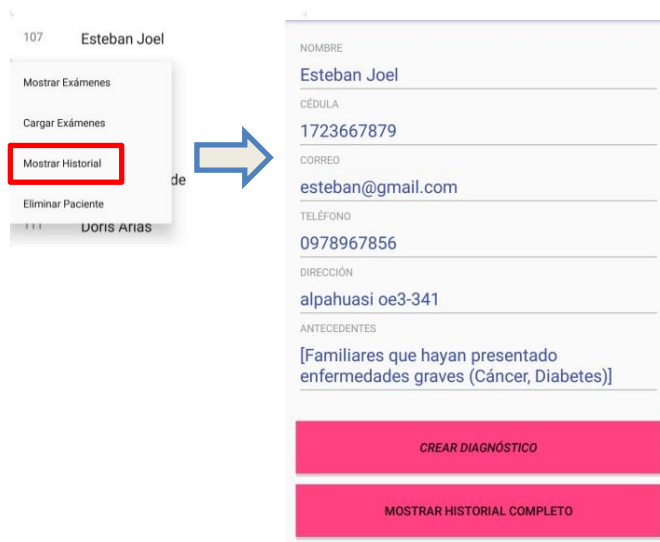


Figura 3.19. Historial completo y creación de diagnósticos.

Un paciente tiene un diagnóstico nuevo cada que se acerca a una visita médica, este diagnóstico debe contener datos como: la descripción de estado de salud actual, la solicitud de exámenes médicos y si es la caso, añadir medicación para el paciente. La

Figura 3.20 muestra el cumplimiento de los requerimientos funcionales: RFM17, RFM18, RFM19.

Figura 3.20. Descripción del estado de salud del paciente.

Una vez creado los diagnósticos éstos pueden ser guardados en un archivo pdf. La Figura 3.21 da cumplimiento al requerimiento funcional: RFM20.

Descripción	Valor
Número de Historial	35
Nombre de Paciente	Blanca Vela
Fecha de Diagnóstico	2018-07-23
Descripción	Presenta molestias en el pie derecho

Figura 3.21. Generación de reporte de diagnóstico.

Cuando el paciente es recién creado no dispondrá de diagnósticos médicos, para informar de esta situación se mostrará un mensaje de información como se muestra en la Figura 3.22.

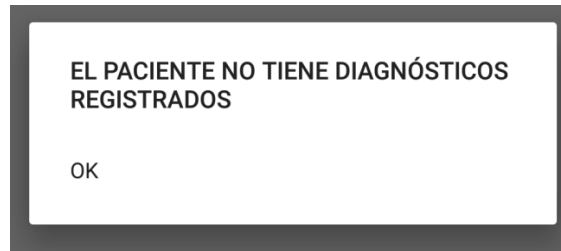


Figura 3.22. Mensaje de notificación que el paciente no presenta diagnósticos.

3.2.6 Pruebas de Validación del Subsistema 06: Gestión de actividades del paciente

En este Subsistema se verificará el funcionamiento de las actividades que podrá realizar un paciente: La Tabla 3.6 muestra los requerimientos funcionales que se debe cumplir para este módulo.

Tabla 3.6. SB06 por comprobar funcionamiento.

SUBSISTEMA	RF ASOCIADO	CASO DE USO	DESCRIPCIÓN
SB06	RFP01	CU06	Loguear paciente.
	RFP01	CU06	Visualizar información personal.
	RFP01	CU06	Visualizar sugerencia de visitas.
	RFP01	CU06	Cargar exámenes médicos.
	RFP01	CU06	Añadir sugerencias.

El paciente una vez que haya sido registrado por un médico podrá acceder al Sistema de Gestión Médica con un nombre de usuario y contraseña (cedula de identidad). La Figura 3.23 muestra el acceso de un paciente. Con esto se da cumplimiento al requerimiento funcional: RFP01.

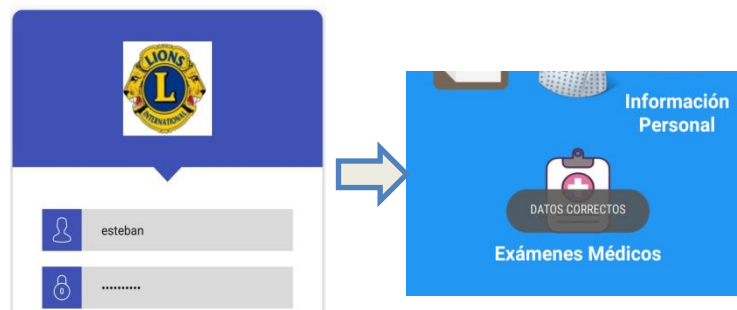


Figura 3.23. Acceso de usuario paciente.

Una vez que el paciente haya accedido al sistema podrá visualizar su información personal. La Figura 3.24 muestra los datos personales del paciente, se recalca que el

paciente no está en capacidad de modificar los datos, dando así cumplimiento al requerimiento funcional: RFP02.



Figura 3.24. Visualización información personal paciente.

El médico puede recomendar al paciente una fecha en la que se puede atender, el paciente podrá observar esta información en su aplicación. La Figura 3.25 muestra un ejemplo de sugerencia que previamente fue registrado en el **Apartado 3.2.4** por el médico dando así cumplimiento al requerimiento funcional: RFP03.

Sistema Gestion Medica		
NOMBRE	FECHA	HORA
David Perez	2018-10-04	15:00:00

Figura 3.25. Sugerencia de cita médica.

El paciente está en la capacidad de subir sus respectivos exámenes médicos, para ello primero debe seleccionar el médico con el cual está siendo tratado, estos exámenes pueden ser tomados con la cámara del dispositivo, la galería o un archivo pdf. La interfaz

que observa el paciente debe ser intuitiva y fácil de utilizar. La Figura 3.26 muestra el cumplimiento del caso de uso: RFP04.



Figura 3.26. Carga de exámenes por parte del paciente.

Los pacientes pueden añadir sus respectivas sugerencias o quejas, estas sugerencias podrán ser visualizadas por el médico que fue atendido. La Figura 3.27 da cumplimiento del requerimiento funcional: RFP05



Figura 3.27. Buzón de sugerencias y quejas.

Habiendo validado cada subsistema por separado, observando así que el software realiza las funciones esperadas y que todos los subsistemas se integran correctamente. Los requerimientos funcionales asociados a cada subsistema permiten verificar que el Sistema de Gestión Médica cumple con las especificaciones y que logra su objetivo. En la Tabla 3.7 se muestra la lista de RF y el cumplimiento de los mismos

Tabla 3.7. Lista de cumplimiento de los requerimientos funcionales.

RF	DESCRIPCIÓN	CUMPLE
RFM01	Registrar médico.	SI
RFM02	Loguear médico.	SI
RFM03	Visualizar información personal médico.	SI
RFM04	Modificar información personal médico.	SI
RFM05	Agregar paciente.	SI
RFM06	Modificar paciente.	SI
RFM07	Eliminar paciente.	SI
RFM08	Buscar paciente.	SI
RFM09	Visualización de los pacientes.	SI
RFM10	Agregar sugerencia de visita.	SI
RFM11	Visualizar agenda.	SI
RFM12	Visualizar sugerencias de la atención.	SI
RFM13	Cargar exámenes médicos.	SI
RFM14	Eliminar exámenes médicos.	SI
RFM15	Crear Historial.	SI
RFM16	Crear diagnóstico.	SI
RFM17	Solicitar exámenes médicos.	SI
RFM18	Añadir descripción (estado de salud).	SI
RFM19	Añadir receta.	SI
RFM20	Generar Reporte.	SI

3.3 Pruebas de Validación de Requerimientos No Funcionales

Se ha realizado una prueba de aceptación del Sistema de Gestión Médica con la colaboración de 10 personas como pacientes y 7 médicos del “Club de Leones Quito”.

Estas pruebas servirán para que el cliente pruebe el software y verifique que cumpla con sus expectativas; esta prueba evalúa la dificultad y tiempo que tomaron en realizar las acciones correspondientes y evaluar si se cumplieron los requerimientos NF, que se encuentran orientado a la calidad del Prototipo, la encuesta fue desarrollada con un formulario de Google para así mostrar los resultados, el formato de las encuestas fueron diseñadas a partir de las especificaciones de requerimientos. En la Figura 3.28 y la Figura 3.29 se puede observar el formato de las encuestas para los participantes.

Sistema de Gestión Médica (Usuario Médico)

Bienvenido y gracias por participar de esta encuesta.

*Obligatorio

1. En una escala del 1 al 5 qué calificación le daría a la aplicación.*

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muy mala	Mala	Buena	Muy Buena	Excelente
2. ¿La presentación de la aplicación es llamativa e intuitiva? *

<input type="radio"/>	Si
<input type="radio"/>	No
3. ¿Pudo crear su perfil de médico sin problemas? *

<input type="radio"/>	Si
<input type="radio"/>	No
4. ¿Pudo eliminar un paciente? *

<input type="radio"/>	Si
<input type="radio"/>	No
5. ¿Pudo crear un diagnóstico, recetar, poner la sugerencia de cita y cargar los exámenes? *

<input type="radio"/>	Si
<input type="radio"/>	No
6. ¿Pudo visualizar la lista de pacientes y el historial de cada uno de ellos? *

<input type="radio"/>	Si
<input type="radio"/>	No
7. El tiempo de espera al realizar una acción como: creación de paciente, visualización y carga de exámenes fue: *

<input type="radio"/>	Lenta (más de 2 min)
<input type="radio"/>	Rápida (menos de 1 min)
9. ¿Cómo evaluaría su experiencia y utilización con esta aplicación? *

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muy difícil	Difícil	Neutro	Fácil	Muy fácil
10. ¿Qué añadiría a la aplicación? *
11. Comentarios

Figura 3.28. Encuesta a Usuario Médico.

Sistema de Gestión Médica (Usuario Paciente)

Bienvenido y gracias por participar de esta encuesta.

*Obligatorio

1. En una escala del 1 al 5 qué calificación le daría a la aplicación.*
Muy mala

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muy mala	Mala	Buena	Muy Buena	Excelente
2. ¿La presentación de la aplicación es llamativa e intuitiva? *

<input type="radio"/>	Si
<input type="radio"/>	No
3. Una vez registrado por su médico, ¿Pudo ingresar con su usuario y cédula de identidad? *

<input type="radio"/>	Si
<input type="radio"/>	No
4. ¿Pudo cargar exámenes médicos? *

<input type="radio"/>	Si
<input type="radio"/>	No
5. ¿Pudo visualizar la sugerencia de cita por parte de su médico? *

<input type="radio"/>	Si
<input type="radio"/>	No
6. ¿El sistema le permitió realizar sugerencias y quejas de la atención médica? *

<input type="radio"/>	Si
<input type="radio"/>	No
7. ¿Pudo visualizar su información personal? *

<input type="radio"/>	Si
<input type="radio"/>	No
8. El tiempo de espera al realizar una acción como la carga de exámenes fue: *

<input type="radio"/>	Lenta (más de 2 min)
<input type="radio"/>	Rápida (menos de 1 min)
9. ¿Cómo evaluaría su experiencia y utilización con esta aplicación? *

1	2	3	4	5
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Muy difícil	Difícil	Neutro	Fácil	Muy fácil
10. ¿Qué añadiría a la aplicación? *

11. Comentarios

Figura 3.29. Encuesta a Usuario Paciente.

La información recolectada total se encuentra en el **ANEXO VII**.

La mencionada prueba fue realizada mediante la recreación del escenario de pruebas de la Figura 1.1 del **Capítulo 1 Apartado 1.2**.

En la Tabla 3.8 se presenta el resultado a la **Pregunta 1** de la encuesta realizada a los médicos. En la misma se puede observar que el 85,7% de los médicos indican que la aplicación es *Excelente*.

Tabla 3.8. Respuesta a la Pregunta 1 del médico.

USUARIO MÉDICO					
PREGUNTA 1	En una escala del 1 al 5 qué calificación le daría a la aplicación				
VALORACION	MUY MALA (1)	MALA (2)	BUENA (3)	MUY BUENA (4)	EXCELENTE (5)
NÚMERO DE RESPUESTAS	0	0	0	1	6
PORCENTAJES	0%	0%	0%	14,3%	85,7%

De la misma manera, en la Tabla 3.9 se presenta el resultado a la **Pregunta 1** de la encuesta realizada a los pacientes. En la misma se puede observar que el 80% de los pacientes indican que la aplicación es *Excelente*.

Tabla 3.9. Respuesta a la pregunta 1 del paciente.

USUARIO PACIENTE					
PREGUNTA	En una escala del 1 al 5 qué calificación le daría a la aplicación				
VALORACIÓN	MUY MALA (1)	MALA (2)	BUENA (3)	MUY BUENA (4)	EXCELENTE (5)
NÚMERO DE RESPUESTAS	0	0	0	2	8
PORCENTAJES	0%	0%	0%	20%	80%

La Tabla 3.10 muestra las preguntas y respuestas que se hizo al médico respecto a su experiencia con el uso del sistema. En base a los resultados obtenidos de la encuesta se puede concluir que los usuarios médicos fueron capaces de: crear su perfil, crear el perfil de un nuevo paciente, eliminar pacientes, crear historiales en los cuales añadían diagnósticos, recetas, la administración de exámenes médicos y sugerencias de citas. Finalmente, la encuesta indica que el médico pudo visualizar la lista de pacientes y el

historial de cada uno. Con base en los resultados se puede concluir que más del 90% de los médicos ha conseguido utilizar la aplicación sin problemas.

Tabla 3.10. Preguntas y respuestas a la encuesta usuario médico.

USUARIO MÉDICO		
Preguntas	Respuestas	
	SI	NO
3. ¿Pudo crear su perfil de médico sin problemas?	7	0
4. ¿Pudo crear el perfil de un nuevo paciente?	7	0
5. ¿Pudo eliminar un paciente?	7	0
6. ¿Pudo crear un diagnóstico, recetar, poner la sugerencia de cita y cargar los exámenes?	6	1
7. ¿Pudo visualizar la lista de pacientes y el historial de cada uno de ellos?	7	0

De la misma manera en la Tabla 3.11 se muestran las preguntas y respuestas que se hizo al paciente respecto a su experiencia con el uso del sistema. En base a los resultados obtenidos de la encuesta se puede concluir que los usuarios pacientes fueron capaces de: *loguearse* al sistema, cargar exámenes médicos, sugerencias de visitas por parte del médico, visualizar su información personal y realizar sugerencias de cómo fue la atención por parte del centro. Con base en los resultados se puede concluir que más del 90% de los pacientes ha conseguido utilizar la aplicación sin problemas.

Tabla 3.11. Preguntas y respuestas a la encuesta usuario paciente.

USUARIO PACIENTE		
Preguntas	Respuestas	
	SI	NO
3. Una vez registrado por su médico, ¿Pudo ingresar con su usuario y cédula de identidad?	10	0
4. ¿Pudo cargar exámenes médicos?	9	1
5. ¿Pudo visualizar la sugerencia de cita por parte de su médico?	9	1
6. ¿El sistema le permitió realizar sugerencias y quejas de la atención médica?	10	0
7. ¿Pudo visualizar su información personal?	10	0

Se verificará la lista de requerimientos NF anteriormente definida en el **Capítulo 2 Apartado 2.1.2** y que se muestran en la Tabla 3.12.

Tabla 3.12 Requerimientos no funcionales.

ETIQUETA	RNF	DESCRIPCIÓN
Presentación	RNF01	Los usuarios deben contar con una interfaz que sea amigable e intuitiva.
Disponibilidad	RNF02	El sistema deberá estar disponible las 24 horas.
Integridad	RNF03	Los datos no deberán ser corruptos por transacciones incompletas o fallidas.
Tiempo de respuesta	RNF04	El tiempo de respuesta deberá ser en pocos segundos. Esto dependerá mucho de la velocidad de acceso a Internet.
Escalabilidad	RNF05	El prototipo debe ser construido en una plataforma que permita el incremento en recursos como almacenamiento, memoria RAM.
Capacidad	RNF06	El sistema deberá ser capaz de soportar mínimo 7 usuarios al mismo tiempo.

3.3.1 RNF01: Presentación

La presentación debe ser atractiva, funcional, intuitiva y amigable [49], esto permite que los usuarios puedan realizar acciones en la aplicación de una manera rápida y sencilla. Por tal razón, en la encuesta realizada se analiza los siguientes parámetros: dificultad, diseño y colores; estos parámetros permitirán evaluar la usabilidad del Sistema de Gestión Médica.

En la Tabla 3.13 se muestran los resultados a la **Pregunta 2** para el usuario médico y el usuario paciente, como se puede observar, en promedio para más del 85% de los encuestados la presentación de la aplicación resulta llamativa.

Tabla 3.13 Respuesta a la Pregunta 2.

PREGUNTA	¿La presentación de la aplicación es llamativa e intuitiva?			
	RESPUESTAS			
USUARIO	SI	PORCENTAJE	NO	PORCENTAJE
MÉDICO	6	85,7%	1	14,3%
PACIENTE	9	90,0%	1	10,0%

3.3.2 RNF02: Disponibilidad

La disponibilidad se define como el grado con que los recursos del sistema están disponibles para su uso por el usuario final, hace referencia a la probabilidad de que el servicio funcione adecuadamente [50]. La implementación del Sistema de Gestión Médica se lo ha realizado utilizando el proveedor de servicios AWS [20], una de las características principales es que se tenga la mayor parte del tiempo el servicio funcionando, es por eso que el *Acuerdo de Nivel de servicios de Amazon* [51] es ofrecer una disponibilidad del 99.99% en todas y cada una de las regiones de Amazon, en base a este dato se da cumplimiento al requerimiento no funcional de *DISPONIBILIDAD*.

3.3.3 RNF03: Integridad

La integridad, supone que la información se mantenga inalterada ante accidentes o intentos maliciosos [52], los datos que sean enviados desde la aplicación no deben estar sujetos a cambios por transacciones incompletas o datos incoherentes. La Figura 3.30 muestra la pantalla donde se registra un paciente nuevo y cómo los datos son guardados en la base de datos, de la misma manera cuando el paciente visualiza su información personal observa los datos que el mismo registró.

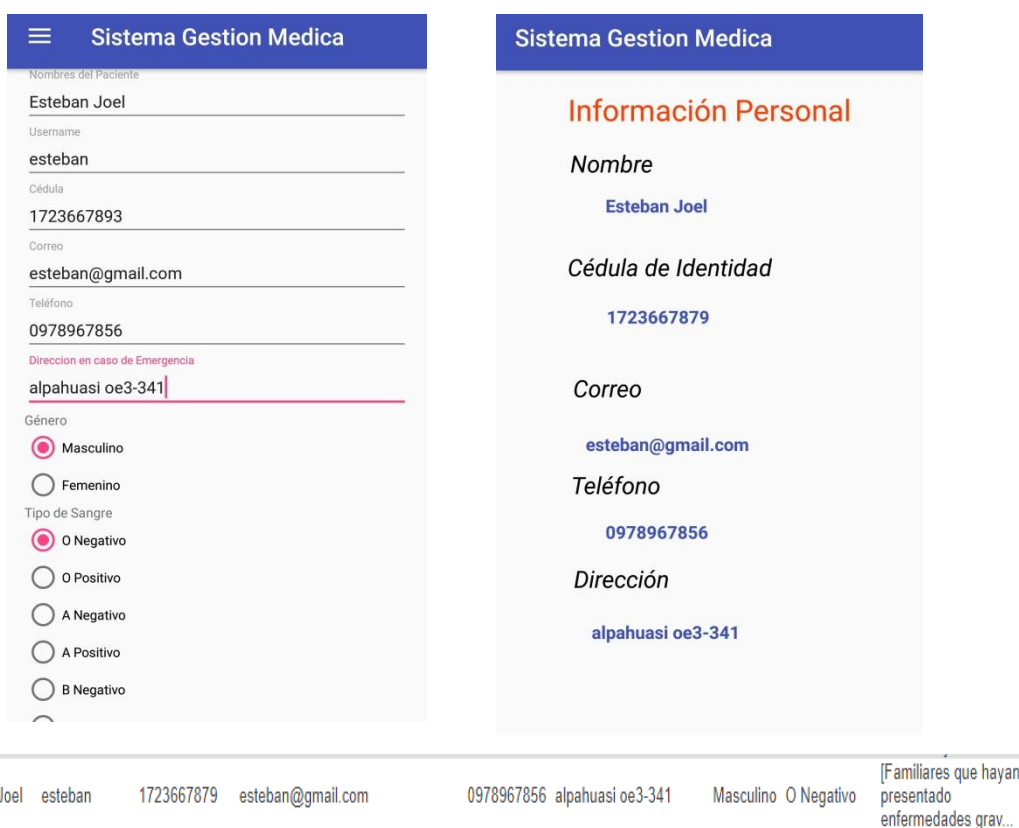


Figura 3.30. Integridad de información personal del paciente.

3.3.4 RNF04: Tiempo de Respuesta

En la Tabla 3.14 se muestran las estadísticas que arrojaron las encuestas donde se dice que los usuarios no tuvieron demora en realizar actividades sobre la aplicación, como se observa más del 85% de los encuestados entre usuarios médicos y pacientes el tiempo de respuesta fue rápida (menos de 1 min). Este es un factor subjetivo que va a depender mucho de la calidad de acceso a internet, modelo de teléfono que tenga el usuario final. Las pruebas se realizaron con datos móviles y con una red Wifi.

Tabla 3.14. Tiempo de realizar una acción.

USUARIO	RESPUESTAS			
	RÁPIDA (MENOS DE 1 MINUTO)	PORCENTAJE	LENTA (MAS DE 2 MINUTOS)	PORCENTAJE
MÉDICO	6	85,7%	1	14,3%
PACIENTE	10	100,0%	0	0,0%

3.3.5 RNF05: Escalabilidad

Escalabilidad es un término para señalar la propiedad de aumentar la capacidad de trabajo o de tamaño de un sistema sin comprometer el funcionamiento y calidad normales del mismo [53]. Como se mencionó en el **Capítulo 2 Apartado 2.1.3** una de las características que brinda el proveedor AWS es la facilidad de crecimiento en recursos como almacenamiento, núcleos de procesador, memoria RAM y velocidad de conexión de datos; esto permite que el usuario no tenga que preocuparse por recursos físicos, por tal razón se considera que el requerimiento de *ESCALABILIDAD* esta cumplido

3.3.6 RNF06: Capacidad

La capacidad es la medida de la cantidad de usuarios que pueden hacer uso de un sistema. En primera instancia el sistema fue planteado para un mínimo de 7 usuarios simultáneos, en el momento de las pruebas se tomó una muestra de 10 pacientes y 7 médicos dando así al cumplimiento del requerimiento no funcional de *CAPACIDAD*.

Así también, se utilizó un software móvil llamado *App Tune-up Kit* [54] que permite analizar cualquier aplicación de Android y permite evaluar el rendimiento de la aplicación

en 3 áreas críticas: potencia, CPU y datos de uso de red. La Tabla 3.15 muestra los datos del rendimiento en varios modelos y marcas cuando el Sistema de Gestión Médica estaba en uso.

Tabla 3.15. Rendimiento del sistema en varios modelos *smartphone*.

MODELO SMARTPHONE	POTENCIA PROMEDIO (mw)	CPU (%)	DATOS DE RED (KB)
SONY XPERIA E5	458	12.7	455,6
SAMSUNG J7 PRIME	562	37.0	525,6
SAMSUNG J5	582	28.0	496,0
HUAWEI	432	14.6	512,3
SAMSUNG S6	584	26.5	625,5

La Figura 3.31 muestra un ejemplo de captura de la aplicación App Tune-up Kit ejecutándose en un *smartphone* de marca SONY, donde se puede visualizar los parámetros de consumo en uso del CPU y consumo de datos de red.

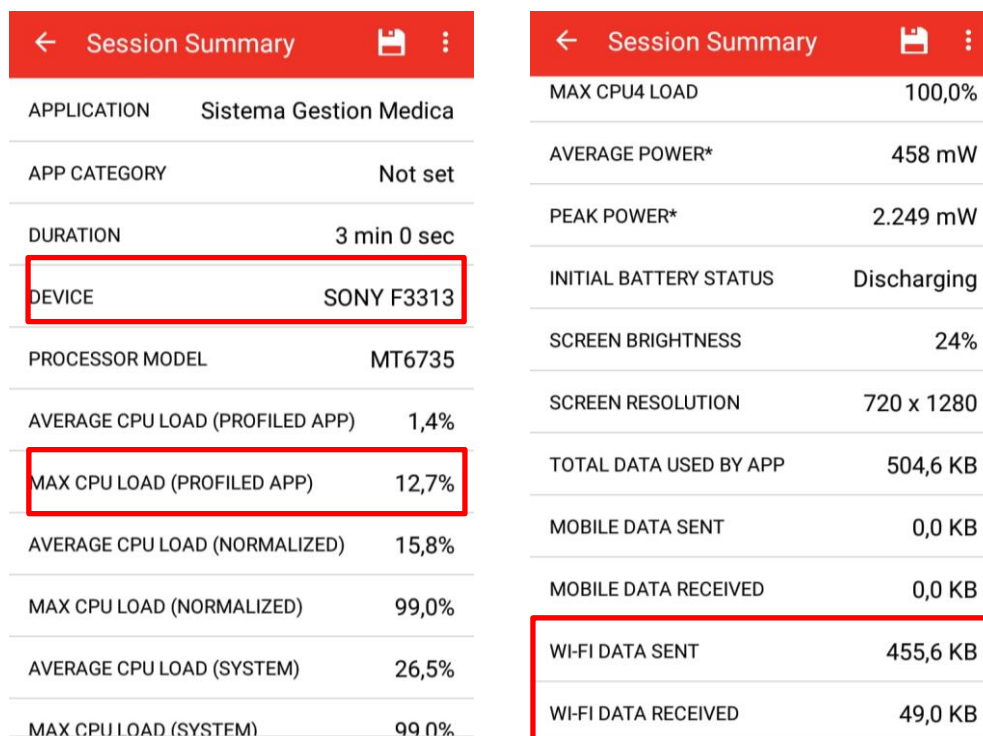


Figura 3.31. Datos de rendimiento en *Smartphone* SONY.

Los requerimientos no funcionales se derivan de factores externos al sistema, permiten la evaluación del servicio desarrollado. Una vez realizada la validación de los requerimientos no funcionales se presenta en la Tabla 3.16 la lista de cumplimiento de los mismos.

Tabla 3.16 Lista de cumplimiento de los Requerimientos No Funcionales.

ETIQUETA	RNF	DESCRIPCIÓN	CUMPLE
Presentación	RNF01	Los usuarios deben contar con una interfaz que sea amigable e intuitiva.	SI
Disponibilidad	RNF02	El sistema deberá estar disponible las 24 horas.	SI
Integridad	RNF03	Los datos no deberán ser corruptos por transacciones incompletas o fallidas.	SI
Tiempo de respuesta	RNF04	El tiempo de respuesta deberá ser en pocos segundos. Esto dependerá mucho de la velocidad de acceso a Internet.	SI
Escalabilidad	RNF05	El prototipo debe ser construido en una plataforma que permita el incremento en recursos como almacenamiento, memoria RAM.	SI
Capacidad	RNF06	El sistema deberá ser capaz de soportar mínimo 7 usuarios al mismo tiempo.	SI

3.4 Tablero Final de Kanban

La Figura 3.32 muestra la organización de las tareas realizadas en el tablero Kanban, en el cual se puede observar que todas las tareas que se han definido para el desarrollo del presente Proyecto de Titulación han sido terminadas. En el tablero en la columna de *Tareas Realizadas* se observa que las tareas planteadas en la etapa de diseño, implementación y pruebas se encuentran finalizadas.

TABLERO FINAL DE KANBAN		
Lista de Tareas	Tareas en Proceso	Tareas Realizadas
		Selección de herramientas a utilizarse
		Realizar encuestas a médicos del área de Traumatología del "Club de Leones Quito"
		Identificación de requerimientos funcionales (RF).
		Identificación de requerimientos no funcionales (RNF).
		Clasificación de los RF por módulos.
		Diseño de los diagramas de casos de uso.
		Diseño del diagrama relacional de base de datos.
		Diseño del diagrama de clases.
		Diseño del servicio Web.
		Diseño de la capa de presentación.
		Diseño de las vistas de la aplicación móvil.
		Creación de una instancia EC2 en AWS.
		Instalación y codificación de WampServer.
		Implementación de la Capa de Datos.
		Implementación de la Capa de Negocio.
		Codificación de las Clases.
		Codificación de los servicios Web.
		Pruebas de funcionamiento al módulo de registro de usuarios (médico y paciente).
		Pruebas de funcionamiento del módulo de registro y visualización de sugerencias de visitas médicas.
		Pruebas de funcionamiento del módulo de registro de sugerencias y quejas por parte de los pacientes.
		Pruebas de funcionamiento del módulo de historiales clínicos.
		Pruebas de funcionamiento del módulo de usuario Paciente
		Pruebas de integración.
		Presentar resultados.

Figura 3.32. Tablero final de Kanban.

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

- Al concluir el presente Proyecto de Titulación se tiene un prototipo de un sistema de aplicación móvil de gestión médica para el área de traumatología en la clínica “Club de Leones Quito”, que aprovecha el incremento del uso de *smartphones* y sus aplicaciones móviles. El prototipo final logra mejorar la forma actual de llevar el registro de historiales clínicos (documentos escritos), solucionando problemas como: escritura ilegible, datos faltantes, formato pobre, dificultad en el acceso y la disponibilidad según se muestra en los datos de las encuestas. Además, con este nuevo prototipo la información se encuentra disponible y respaldada a través de Internet mediante el proveedor *cloud* AWS.
- Durante la elaboración de este Proyecto de Titulación se utilizó la metodología KANBAN permitiendo organizar y mantener en orden las tareas, mejorar el proceso de desarrollo, y limitar el trabajo en curso en cada fase evitando atrasos en la realización del prototipo. Esta metodología permitió que en cada una de las capas: Datos, Negocio y Presentación se pueda dividir en tareas más pequeñas. Las tareas por mencionar algunas han sido selección de un proveedor de *Cloud Computing*, análisis de requerimientos, diseño de diagramas Entidad-Relación, clases, etc.
- El presente Proyecto de Titulación permitió analizar diferentes fundamentos permitiendo aprender y desarrollar habilidades en el área de Diseño de Software, ya que se ha manejado diferentes herramientas, tecnologías y conceptos como son: Arquitectura en Capas, Programación en Servicios, *Cloud Computing*, Android Studio, MySQL, NetBeans, WampServer, etc. haciendo que se afiance conocimientos en el área de programación y base de datos. Todos los temas mencionados han sido utilizados en la elaboración de este Proyecto.
- Mediante las encuestas se pudo realizar un correcto análisis de requerimientos funcionales y no funcionales los cuales permiten saber las necesidades que tiene el usuario final (médicos y pacientes) y, además permite al desarrollador tener un panorama más claro en el proceso de diseño, como por ejemplo el realizar los respectivos diagramas en cada una de las capas: el diagrama entidad relación, diagramas de caso de uso y clases. Además, estos requerimientos ayudan también para el diseño de la interfaz gráfica que llevará la aplicación logrando satisfacción del usuario.

- Se hace notar que tanto la Capa de Datos y Negocio se encuentran implementados en el entorno de la nube, permitiendo independencia de la Capa de Presentación. Finalmente, la Capa de Presentación se encuentra alojada en el dispositivo de cada usuario, permitiendo así la separación de las partes que componen el sistema total.
- Las pruebas de aceptación del Sistema de Gestión Médica verificaron que se cumplió con los requerimientos levantados. Además, de las encuestas y de los resultados obtenidos se puede concluir que más del 90% de los usuarios se sintieron cómodos al usar la aplicación.

4.2 Recomendaciones

- La buena práctica del uso de una metodología permite al desarrollador llevar de manera más ordenada la evolución del proyecto y garantizar los resultados en tiempos definidos. Se recomienda escoger la metodología que mejor se acople a los requerimientos del proyecto previamente al inicio del mismo. El uso de KANBAN en este proyecto ha facilitado el proceso de desarrollo.
- El usuario final cumple un papel importante en el desarrollo de cualquier sistema de software, por lo cual es importante tener en cuenta la importancia de tener reuniones periódicas con las personas involucradas que han solicitado el desarrollo del software, pues al no hacerlo puede que el cliente no se encuentre satisfecho con el producto final. Esto provoca demoras en la entrega del software, pues, los nuevos cambios pueden implicar cambios en cada una de las capas: datos, negocio o presentación.
- Dado que el desarrollo de aplicaciones en la actualidad ha tomado un impulso fuerte, entonces se podría considerar importante incentivar en la carrera en Ingeniería en Electrónica y Redes de Información materias en las cuales se enseñe sobre el desarrollo de aplicaciones móviles para que los estudiantes puedan tener otra opción al momento de buscar trabajo o emprender.
- Para un sistema final donde se necesite de mayor cantidad de usuarios, que ya no sea solo un área sino toda una entidad de salud como por ejemplo un hospital, es evidente que el uso del servidor será más alto, por tanto se necesitará de mayor capacidad de disco duro, procesador, ancho de banda y memoria RAM. Estos datos pueden ser calculados mediante el número de usuarios simultáneos, tamaño de archivos a enviar. Los archivos de texto no son de gran tamaño.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] MV - Líder en Sistemas de Gestión de Salud, «Salud Digital: aplicaciones revolucionaron relación médico-paciente». [En línea]. Disponible en: <http://www.mv.com.br/es/blog/salud-digital--aplicaciones-revolucionaron-relacion-medico-paciente>. [Accedido: 20-ene-2018].
- [2] T. Harris, «CLOUD COMPUTING – An Overview». [En línea]. Disponible en: <http://www.thbs.com/downloads/Cloud-Computing-Overview.pdf>. [Accedido: 20-dic-2017].
- [3] Página Oficial, «CLUB DE LEONES QUITO CENTRAL», 31-dic-2017. [En línea]. Disponible en: <http://www.cmcquitocentral.org/historia.php>. [Accedido: 31-dic-2017].
- [4] S. A. Betancourt Kaltsidou, A. M. Llinás Volpe, y C. P. Novoa Ramírez, «Aplicaciones (APPS), una innovadora tendencia en salud: desarrollo de una aplicación móvil para manejo de sucesos críticos durante la cirugía ortopédica y traumatológica», vol. 31, 01-dic-2017.
- [5] «ICUC: La app uruguaya para medicina interactiva», *Cuti*, 16-ene-2018. [En línea]. Disponible en: <http://www.cuti.org.uy/novedades/317-icuc-la-app-uruguaya-para-medicina-interactiva>. [Accedido: 16-ene-2018].
- [6] G. Ferriero *et al.*, «Reliability of a smartphone-based goniometer for knee joint goniometry», *Int. J. Rehabil. Res. Int. Z. Rehabil. Rev. Int. Rech. Readaptation*, vol. 36, n.º 2, pp. 146-151, jun. 2013.
- [7] J. G. Enriquez y S. I. Casas, «USABILIDAD EN APLICACIONES MÓVILES», vol. 5, n.º 2, p. 23, 2013.
- [8] J. Navarro, «La evolución de los smartphones», 25-ene-2018. [En línea]. Disponible en: <http://histinf.blogs.upv.es/files/2012/12/Evoluci%C3%B3n-de-los-Smartphones-Blog-HDI.pdf>. [Accedido: 25-ene-2018].
- [9] Página Oficial, «Google Play». [En línea]. Disponible en: <https://play.google.com/store?hl=es>. [Accedido: 16-sep-2018].
- [10] Página Oficial, «App Store», *Apple (América Latina)*. [En línea]. Disponible en: <https://www.apple.com/la/ios/app-store/>. [Accedido: 16-sep-2018].
- [11] S. Carrasco Usano, «Análisis de la aplicación de la tecnología móvil en las empresas», Universidad Politécnica de Valencia, Valencia, 2015.

- [12] A. Moscaritolo, «El 99.6% del mercado móvil le pertenece a Android y iOS», *PCMag Latam*, 17-feb-2017. [En línea]. Disponible en: <http://latam.pcmag.com/sistemas-operativos-moviles/18490/news/el-996-del-mercado-movil-le-pertenece-a-android-y-ios>. [Accedido: 01-ene-2018].
- [13] Organización Mundial de la Salud, «mSalud: uso de las tecnologías móviles inalámbricas en la salud pública», OMS, may 2016.
- [14] M. T. Sanchez Rodríguez, S. Collado Vásquez, P. Marín Casas, y R. Cano de la Cuerda, «Apps en neurorrehabilitación. Una revisión sistemática de aplicaciones móviles», España, p. 14, oct-2015.
- [15] J. Cepeda, «Tipos y utilidades de las aplicaciones móviles de salud», *Salud Conectada*, 16-ene-2018. [En línea]. Disponible en: <https://saludconectada.com/1-4-tipos-utilidades-las-aplicaciones-moviles-salud/>. [Accedido: 16-ene-2018].
- [16] T. G. Peter Mell, «The NIST Definition of Cloud Computing», *Septiembre 2011*, n.º 800-145, p. 7, ene. 2018.
- [17] O. A. Mejía, «Computación en la nube», 25-ene-2018. [En línea]. Disponible en: <http://www2.izt.uam.mx/newpage/contactos/anterior/n80ne/nube.pdf>. [Accedido: 25-ene-2018].
- [18] Página Oficial, «Cloud Computing, servicios de alojamiento y APIs de Google Cloud», *Google Cloud Platform*, 30-ene-2018. [En línea]. Disponible en: <https://cloud.google.com/?hl=es>. [Accedido: 30-ene-2018].
- [19] Página Oficial, «Microsoft Azure: plataforma y servicios de informática en la nube». [En línea]. Disponible en: <https://azure.microsoft.com/es-es/>. [Accedido: 15-may-2018].
- [20] Página Oficial, «AWS | Cloud Computing - Servicios de informática en la nube», *Amazon Web Services, Inc.*, 24-ene-2018. [En línea]. Disponible en: <https://aws.amazon.com/es/>. [Accedido: 24-ene-2018].
- [21] Página Oficial, «Database | Cloud Database | Oracle». [En línea]. Disponible en: <https://www.oracle.com/database/index.html>. [Accedido: 15-may-2018].
- [22] Página Oficial, «IBM Cloud», 14-may-2015. [En línea]. Disponible en: <https://www.ibm.com/cloud/>. [Accedido: 15-may-2018].

- [23] R. H. Dueñas, «Los 5 principales proveedores Cloud: #1 Microsoft, #2 Amazon, #3 IBM, #4 Salesforce, #5 SAP. Según Forbes 2017.», 09-nov-2017. [En línea]. Disponible en: <https://www.linkedin.com/pulse/los-5-principales-proveedores-cloud-1-microsoft-2-3-4-rayner>. [Accedido: 20-feb-2018].
- [24] S. Goyal, «Public vs Private vs Hybrid vs Community - Cloud Computing: A Critical Review», feb. 2014.
- [25] Advisera, «¿Qué es norma ISO 27001?», *27001Academy*. [En línea]. Disponible en: <https://advisera.com/27001academy/es/que-es-iso-27001/>. [Accedido: 07-mar-2018].
- [26] Página Oficial, «Web Hostin - CNT», *Internet | Corporación Nacional de Telecomunicaciones*, 01-ene-2018. [En línea]. Disponible en: <https://www.cnt.gob.ec/internet/plan-corporativo/web-hosting/>. [Accedido: 01-ene-2018].
- [27] S. D. Bijwe y Prof. P. L. Ramteke, «Database in Cloud Computing - Database-as-a Service (DBaaS) with its Challenges», vol. 4, pp. 73-79, feb. 2015.
- [28] M. Besteiro y M. Rodriguez, «Web Services», 14-mar-2018. [En línea]. Disponible en: <http://www.ehu.es/mrodriguez/archivos/csharp/pdf/ServiciosWeb/WebServices.pdf>.
- [29] C. G. Arcos, «Los Web Services y Características de Calidad», 14-mar-2018. [En línea]. Disponible en: http://www.unilibre.edu.co/revistaavances/avances_10/r10_art7.pdf. [Accedido: 14-mar-2018].
- [30] Página Oficial, «Introducción a JSON», 15-mar-2018. [En línea]. Disponible en: <https://www.json.org/json-es.html>. [Accedido: 15-mar-2018].
- [31] Página Oficial, «Conoce Android Studio | Android Studio», 16-ene-2018. [En línea]. Disponible en: <https://developer.android.com/studio/intro/index.html?hl=es-419>. [Accedido: 16-ene-2018].
- [32] Página Oficial, «NetBeans IDE - Overview», 16-mar-2018. [En línea]. Disponible en: <https://netbeans.org/features/index.html>. [Accedido: 16-mar-2018].
- [33] Página Oficial, «WampServer», *WampServer*, 16-mar-2018. [En línea]. Disponible en: <http://www.wampserver.com/>. [Accedido: 16-mar-2018].
- [34] B. Marcos, «El Kanban», Universidad Oberta de Catalunya, p. 35, 21-feb-2018.
- [35] H. Kniberg y M. Skarin, «Kanban y Scrum – obteniendo lo mejor de ambos», 21-feb-2018. [En línea]. Disponible en:

http://www.proyectalis.com/documentos/KanbanVsScrum_Castellano_FINAL-printed.pdf.
[Accedido: 21-feb-2018].

[36] J. S. Pinto de los Ríos, «Implementación del método Kanban en las empresas constructoras pequeñas y medianas en la ejecución de un proyecto en Colombia.», Universidad Politécnica de Valencia, Valencia, 2015.

[37] Ulf Eriksson, «Functional Requirements vs Non Functional Requirements», *ReQtest*, 05-abr-2012. .

[38] Goyes, Javier y Aimacaña Quilumba, Doris Eulalia, «Análisis, diseño e implementación de un Data Mart Académico usando tecnología de BI para la Facultad de Ingeniería, Ciencias Físicas y Matemática.», Universidad Central del Ecuador, Quito, 2013.

[39] UDESIGN, «Criterios de evaluación cuadrante mágico de gartner.», 05-mar-2018. [En línea]. Disponible en: <https://incared.net/criterios-de-evaluacion-cuadrante-magico-de-gartner/>. [Accedido: 05-mar-2018].

[40] Página Oficial, «AWS nombrado líder en el informe de Magic Quadrant en referencia a la infraestructura como servicio (IaaS) por sexto año consecutivo», 05-mar-2018. [En línea]. Disponible en: <https://aws.amazon.com/es/resources/gartner-2016-mq-learn-more/>. [Accedido: 05-mar-2018].

[41] A. López Gaona, «El modelo Entidad-Relación», Facultad de Ciencias, UNAM, 07-mar-2018.

[42] Página Oficial, «Diagramas de Flujo Online», *Lucidchart*, 20-abr-2018. [En línea]. Disponible en: <https://www.lucidchart.com/pages/es>. [Accedido: 10-jun-2018].

[43] Sonali, «Advantages of PHP Programming». [En línea]. Disponible en: <https://www.eukhost.com/blog/webhosting/advantages-of-php-programming/>. [Accedido: 20-jul-2018].

[44] Página Oficial, «Home - Pencil Project». [En línea]. Disponible en: <https://pencil.evolus.vn/>. [Accedido: 11-jun-2018].

[45] Página Oficial, «AWS Management Console». [En línea]. Disponible en: <https://console.aws.amazon.com/console/home?region=us-east-1>. [Accedido: 12-jun-2018].

- [46] Página Oficial, «Cloud Compute Products – Amazon Web Services (AWS)», *Amazon Web Services, Inc.* [En línea]. Disponible en: <https://aws.amazon.com/products/compute/>. [Accedido: 12-jun-2018].
- [47] Página Oficial, «NetBeans IDE Download». [En línea]. Disponible en: <https://netbeans.org/downloads/>. [Accedido: 23-jul-2018].
- [48] Página Oficial, «Material Design for Android», *Android Developers*. [En línea]. Disponible en: <https://developer.android.com/guide/topics/ui/look-and-feel/>. [Accedido: 24-jun-2018].
- [49] J. Cuello, «Capítulo 8: Diseño visual – Diseñando apps para móviles», *Designing mobile apps*. [En línea]. Disponible en: <http://appdesignbook.com/es/contenidos/diseño-visual-apps-nativas/>. [Accedido: 17-sep-2018].
- [50] ISOTools Excellence, «Confidencialidad, integridad y disponibilidad en los SG-SSI», *PMG SSI - ISO 27001*, 01-feb-2018. [En línea]. Disponible en: <https://www.pmg-ssi.com/2018/02/confidencialidad-integridad-y-disponibilidad/>. [Accedido: 23-sep-2018].
- [51] Página Oficial, «Amazon Compute Service Level Agreement», *Amazon Web Services, Inc.* [En línea]. Disponible en: <https://aws.amazon.com/compute/sla/>. [Accedido: 06-jul-2018].
- [52] Métricas de un proyecto, «5.2 FACTORES DE CALIDAD DEL SOFTWARE | SELECCIÓN DE TÉCNICAS DE INGENIERÍA DE SOFTWARE». [En línea]. Disponible en: https://virtual.itca.edu.sv/Mediadores/stis/52___factores_de_calidad_del_software.html. [Accedido: 17-sep-2018].
- [53] L. Castro, «¿A qué se refieren con eso de escalabilidad?», *aboutespanol*. [En línea]. Disponible en: <https://www.aboutespanol.com/que-es-escalabilidad-157635>. [Accedido: 23-sep-2018].
- [54] Qualcomm Developer Network, «App Tune-up Kit», *Qualcomm Developer Network*. [En línea]. Disponible en: <https://developer.qualcomm.com/software/app-tune-up-kit>. [Accedido: 12-ago-2018].

6. ANEXOS

ANEXO I. Encuestas realizadas a los médicos y resultados de las mismas.

ANEXO II. Descripción y Codificación de la Base de Datos.

ANEXO III. Script para la codificación de la Base de Datos.

ANEXO IV. Scripts para la codificación de las Clases.

ANEXO V. Scripts para la codificación de los Servicios Web.

ANEXO VI. Scripts para la codificación de las Interfaces Gráficas y código en Android Studio.

ANEXO VII. Encuestas a colaboradores del prototipo

NOTA: Los anexos se encuentran adjuntos en el CD.

ORDEN DE EMPASTADO