

**UNIVERSIDAD DE LOS ANDES  
FACULTAD DE INGENIERÍA  
DIVISIÓN DE ESTUDIOS DE POSTGRADO  
POSTGRADO EN COMPUTACIÓN**



**Arquitectura para el despliegue tridimensional en dispositivos móviles de  
datos generados por tomógrafos.**

Autor: Carlos Javier Bahena Reyes  
Tutor: MSc. Domingo Hernández

Trabajo de grado presentado ante la Ilustre Universidad de Los Andes como requisito  
parcial para optar al grado de *Magister Scientiae en Computación*

Mérida, Septiembre 2016

## RESUMEN

El presente trabajo de grado presenta una propuesta básica de una arquitectura para el soporte del despliegue tridimensional de imágenes generadas por tomógrafos y el desarrollo de un prototipo que cumpla los requisitos mínimos para cada elemento compuesto por dicha arquitectura, permitiendo la visualización de archivos 3D en formato Stereolithography (STL), la arquitectura comienza por la captura de las imágenes médicas que pueden ser obtenidas de diferentes fuentes, como la Tomografía computarizada, ultrasonido, resonancia magnética, entre otros. Estas imágenes se encuentran en formato DICOM y se envían al Sistema de Archivo y Comunicación de Imagen (PACS) para su posterior procesamiento tridimensional.

Adicionalmente se cuenta con un sistema web desarrollado en PHP que se conecta a un servidor web Apache 2.4.10 que sirve como plataforma para realizar las conexiones bidireccionales con el cliente mediante el protocolo HTTP. A su vez el sistema web se conecta a la base de datos centralizada Mysql versión 5.6.21 y un prototipo para el despliegue tridimensional que se lleva a cabo bajo plataforma Android 4.2.2 para dispositivos móviles mediante una interfaz de programación de aplicaciones (API) desarrollada en PHP 5.6.3. Cada elemento en conjunto correspondiente a la plataforma propuesta tiene como aporte científico que es totalmente escalable, en código libre, ligero, abierto al uso de otras tecnologías móviles o no para el despliegue de los modelos de estudio y cubre cada etapa del proceso de captura de datos hasta el diagnóstico de los resultados por parte del especialista médico.

El trabajo de grado se desarrolla usando la metodología en espiral Boehm, el cual se lleva a cabo mediante un enfoque evolutivo, creando prototipos funcionales para así generar el prototipo básico para cumplir con el objetivo general, además que puede adaptarse y aplicarse a lo largo de la vida del software de computadora.

**Palabras Clave:** Reconstrucción, Tridimensional, Arquitectura de visualización de datos gráficos, despliegue tridimensional en dispositivos móviles.

## ÍNDICE GENERAL

RESUMEN .....	ii
Índice de Figuras .....	vi
Índice de Tablas.....	viii
Índice de Código de Programación .....	ix
<b>CAPÍTULO I</b> .....	<b>1</b>
<b>GENERALIDADES</b> .....	<b>1</b>
1.1    Introducción. ....	1
1.2    Antecedentes .....	3
1.3    Planteamiento del problema. ....	10
1.4    Objetivos. ....	12
1.4.1    Objetivo General. ....	12
1.4.2    Objetivos Específicos. ....	12
1.5    Alcance.....	13
1.6    Justificación.....	13
1.7    Organización del Documento. ....	15
<b>CAPÍTULO II</b> .....	<b>16</b>
<b>MARCO TEÓRICO</b> .....	<b>16</b>
2.1    Elementos conceptuales.....	16
2.1.1    Tomografía. ....	16
2.1.2    Tipos de cortes en la tomografía computarizada.....	17
2.1.3    Tipos de Tomógrafos.....	18
2.1.4    Sistema de Archivado y Transmisión de Imágenes (PACS).....	18
2.1.5    Estándares para imágenes médicas .....	20
2.1.5.1    Estándar DICOM.....	21
2.1.5.2    Formato de una archivo DICOM .....	23
2.1.5.3    Estructura DICOMDIR.....	25
2.2    Formato de archivos STL .....	25
2.2.1    STL binario .....	26
2.2.2    STL ASCII .....	27
2.3    Servidor web .....	27
2.4    Repositorio de modelos 3D .....	28
2.5    Plataforma de interconexión.....	28
2.6    Dispositivos Móviles.....	29

2.7 Contribución Antecedentes .....	29
<b>CAPÍTULO III .....</b>	<b>31</b>
<b>METODOLOGÍA.....</b>	<b>31</b>
3.1 Tipo de investigación .....	31
3.2 Diseño de la investigación.....	32
3.3 Técnicas e instrumentos .....	32
3.4 Metodología de desarrollo .....	33
<b>CAPÍTULO IV .....</b>	<b>36</b>
<b>DISEÑO DE LA ARQUITECTURA.....</b>	<b>36</b>
4.1 Determinación de los objetivos .....	36
4.2 Fase de diseño.....	37
4.2.1 Conceptualización. ....	37
4.2.1.1 Requisitos Funcionales: .....	37
4.2.1.2 Casos de uso .....	39
4.2.1.3 Diagrama de actividades:.....	45
4.2.1.4 Diagrama de Secuencia para el prototipo móvil .....	48
4.2.1.5 Repositorio de modelos .....	51
4.2.1.6 Diagrama de paquetes del prototipo propuesto .....	52
4.2.1.7 Diagrama de clases del prototipo propuesto .....	53
4.2.1.8 Visión Arquitectónica del Sistema.....	54
4.2.1.9 Diagrama de Despliegue.....	55
4.2.1.10 Requisitos no funcionales: .....	56
4.2.1.11 Especificaciones: .....	57
<b>CAPÍTULO V .....</b>	<b>61</b>
<b>IMPLEMENTACIÓN Y FUNCIONAMIENTO DE LA ARQUITECTURA PROPUESTA</b> .....	<b>61</b>
5.1 Arquitectura Propuesta .....	61
5.2 Repositorio de modelos .....	62
5.3 Cliente web .....	63
5.4 Sistema web .....	63
5.5 API PHP.....	67
5.5.1 Estructura API PHP .....	67
5.5.2 Solicitud y Respuesta API PHP .....	69
5.6 Prototipo de Dispositivo Móvil .....	74

5.6.1 Entorno de software.....	74
5.6.2 Funcionamiento del prototipo.....	75
5.6.3 Pantallas y formularios del prototipo.....	75
5.6.4 ConFiguración para el funcionamiento del prototipo en la red de la arquitectura.....	76
5.7 Renderización de la escena 3D.....	78
<b>CAPÍTULO VI.....</b>	<b>80</b>
<b>PRUEBAS FUNCIONALES Y DE RENDIMIENTO .....</b>	<b>80</b>
6.1 Pruebas funcionales.....	80
6.2 Pruebas de rendimiento .....	82
6.2.1 Especificaciones red inalámbrica para prueba rendimiento .....	83
6.2.2 Especificaciones del dispositivo móvil para prueba de rendimiento.....	83
6.2.3 Entorno y estudios de rendimiento .....	83
<b>CAPÍTULO VII.....</b>	<b>93</b>
<b>CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>93</b>
7.1 Conclusiones .....	93
7.2 Recomendaciones.....	95
<b>REFERENCIAS .....</b>	<b>97</b>
<b>ANEXOS .....</b>	<b>100</b>
Diagrama de clases para el prototipo móvil.....	100

## Índice de Figuras

Figura 1: Experimentación del flujo de trabajo de proyecto (Tomado de Fermi, Supramaniam, Kuo, Ammar, Anil y Rajeswari, 2012) .....	4
Figura 2: Diagrama de la arquitectura del sistema propuesto por Fermi y otros (Fermi, Supramaniam, Kuo, Ammar, Anil y Rajeswari, 2012) .....	5
Figura 3: Arquitectura para el visor DICOM en X3D (Tomado de Ojog y Arias, 2012) .....	7
Figura 4: Arquitectura para el despliegue 3D (Tomado de Ramírez y Coto, 2012) .....	9
Figura 5: Diagrama de tomografía (Tomado de Franco, 2011) .....	17
Figura 6: Tipos de corte de las tomografías (Tomado de Franco, 2011) .....	17
Figura 7: Diagramas de sistemas PACS (Tomado de Franco, 2011) .....	20
Figura 8: Estándar DICOM (Tomado de Franco, 2011) .....	23
Figura 9: Estructura de Archivo DICOM (Tomado de Martínez, 2006) .....	24
Figura 10: Estructura archivo DICOM (Tomado de Martínez, 2006) .....	24
Figura 11: Estructura Archivos DICOM (Tomado de Martínez, 2016) .....	25
Figura 12: Modelo en espiral de Boehm (Tomado de Santos, 2014) .....	35
Figura 13: Diagrama de Actores .....	39
Figura 14: Caso de uso radiólogo .....	40
Figura 15: Caso de uso sistema web .....	41
Figura 16: Caso de uso prototipo móvil .....	43
Figura 17: Diagrama de Actividades Unidad Radiología .....	46
Figura 18: Diagrama de Actividades Sistema Web .....	47
Figura 19: Diagrama de Actividades prototipo móvil .....	47
Figura 20: Diagrama de secuencia inicio de sesión .....	48
Figura 21: Diagrama de secuencia cambiar contraseña .....	49
Figura 22: Diagrama de secuencia para cargar la Bandeja de Exámenes .....	50
Figura 23: Diagrama de secuencia para desplegar modelo 3D .....	50
Figura 24: Diagrama de secuencia cerrar sesión .....	51
Figura 25: Diagrama de paquetes prototipo propuesto .....	53
Figura 26: Diagrama de clases del prototipo .....	54
Figura 27: Visión arquitectónica del sistema .....	55
Figura 28: Diagrama de Despliegue .....	56
Figura 29: Modelo entidad relación Base de datos propuesta .....	60
Figura 30: Arquitectura propuesta .....	61
Figura 31: Organización del repositorio de modelos 3D .....	62
Figura 32: Inicio de sesión del sistema web .....	64
Figura 33: Menú principal sistema web .....	64
Figura 34: Menú herramientas sistema web .....	65
Figura 35: Formulario carga del examen en sistema web .....	66
Figura 36: Formulario registro de usuarios en sistema web .....	67
Figura 37: Diagrama estructura API PHP .....	69
Figura 38: Vista del Prototipo. (a) Inicio de Sesión. (b) Cambio de contraseña. (c) Bandeja de Exámenes. (d) Descarga de modelo 3D. (e) Carga de archivo STL. (f) Despliegue 3D del modelo. (g) Preferencias Despliegue 3D .....	76
Figura 39: Despliegue del listado bandeja de exámenes .....	81
Figura 40: Despliegue 3D examen médico .....	82
Figura 41: Visualización de modelos de prueba. (a) Modelo Leg.stl (b) Modelo Spine_Thoracic_Lumbar_half.stl (c) Modelo Mandible.stl .....	84

Figura 42: <i>Consumo de memoria pantalla inicio prototipo</i> .....	85
Figura 43: <i>Consumo de memoria bandeja entrada del prototipo</i> .....	85
Figura 44: <i>Consumo de memoria modelo Leg.stl</i> .....	86
Figura 45: <i>Consumo de memoria modelo Spine_Thoracic_Lumbar_half.stl</i> .....	87
Figura 46: <i>Consumo de memoria modelo Mandible.stl</i> .....	88
Figura 47: <i>Velocidad de descarga modelo Spine_Thoracic_Lumbar_half.stl</i> .....	88
Figura 48: <i>Velocidad de descarga modelo Mandible.stl</i> .....	89
Figura 49: <i>Porcentaje de uso de CPU Spine_Thoracic_Lumbar_half.stl</i> .....	89
Figura 50: <i>Porcentaje de uso de CPU Leg.stl</i> .....	90
Figura 51: <i>Porcentaje de uso de CPU Mandible.stl</i> .....	90
Figura 52: <i>Frecuencia de reloj para el modelo Leg.stl</i> .....	91
Figura 53: <i>Línea de tiempo métodos prototipo</i> .....	92
Figura 54: <i>Línea de tiempo descarga archivo Leg.stl</i> .....	92
Figura 55: <i>Diagrama de clases prototipo móvil</i> .....	100

## Índice de Tablas

Tabla 1. <i>Tabla comparativa de arquitecturas antecedentes</i> .....	10
Tabla 2. <i>Contribución de los antecedentes</i> .....	30
Tabla 3. <i>Requisitos Funcionales</i> .....	39
Tabla 4. <i>Descripción Caso de Uso Unidad de Radiología</i> .....	41
Tabla 5. <i>Descripción Caso de Uso Sistema Web</i> .....	43
Tabla 6: <i>Descripción Caso de Uso Prototipo Móvil</i> .....	45
Tabla 7: <i>Entrada/Salida proceso inicio de sesión</i> . .....	70
Tabla 8: <i>Definición inicio de sesión correcta o incorrecta en JSON</i> .....	71
Tabla 9: <i>Entrada/Salida proceso Consultar Bandeja de Exámenes</i> . .....	71
Tabla 10: <i>Definición inicio de sesión correcta o incorrecta en JSON</i> .....	72
Tabla 11: <i>Entrada/Salida proceso cambiar contraseña</i> . .....	73
Tabla 12: <i>Definición inicio de sesión correcta o incorrecta en JSON</i> .....	74
Tabla 13: <i>Entrada/Salida proceso cerrar sesión</i> .....	74
Tabla 14: <i>Cuadro de prueba de funcionamiento para listado de estudiantes</i> . .....	80
Tabla 15: <i>Cuadro de prueba de funcionamiento para listado de estudiantes</i> . .....	82
Tabla 16: <i>Especificaciones del dispositivo móvil de prueba (Tomado de Huawei Technologies Co, 2014)</i> .....	83
Tabla 17: <i>Características de los modelos de prueba</i> .....	84
Tabla 18: <i>Detalle consumo de memoria modelo Spine_Thoracic_Lumbar_half.stl</i> .....	86
Tabla 19: <i>Detalle consumo de memoria modelo Spine_Thoracic_Lumbar_half.stl</i> .....	87



## Índice de Código de Programación

<i>Código de Programación 1: Clase ServerSetting</i> .....	77
<i>Código de Programación 2: Archivo DB_config.php</i> .....	77
<i>Código de Programación 3. Lectura STL ASCII (STLObject.java)</i> .....	79

# CAPÍTULO I

## GENERALIDADES

### 1.1 Introducción.

El hombre desde siempre se ha valido de herramientas e instrumentos para hacer más fácil sus tareas, es por eso que actualmente en una sociedad tan moderna y compleja en cuanto a avances tecnológicos, es indispensable el uso de la tecnología si se quiere estar a la par de un mundo tan cambiante.

Actualmente existen sistemas para la captura, gestión, tratamiento y almacenamiento de imágenes médicas, arqueología, biología, oceanografía, ciencia de los materiales y otras ciencias gracias al surgimiento de la tomografía computarizada, el cual ha incrementado la cantidad de estudios y de equipos médicos capaces de adquirir, almacenar y visualizar imágenes con la finalidad de evitar el uso de películas, discos compactos (CD, por sus siglas en inglés) o documentos en papel (Franco, 2011).

Es vital el uso de las tecnologías de la información (TI) para el análisis de imágenes médicas junto los sistemas de visualización se han convertido en una herramienta muy importante para valorar múltiples padecimientos y alteraciones corporales como fracturas, hemorragias internas, tumores o infecciones en los distintos órganos, posibilitando la especificación de cada situación de una forma muy acertada sirviendo a los médicos como una guía para la realización de intervenciones mínimamente invasivas, toma de biopsias, drenaje de abscesos, reduciendo la necesidad de intervenciones quirúrgicas.

Hoy en día gracias a las capacidades de cómputo de los dispositivos móviles como teléfonos inteligentes y tabletas existen diversas investigaciones sobre su uso en el procesamiento de imágenes médicas que permiten cada día más migran de sistemas sin conexión a aplicaciones

conectadas en la red disponibles en todo el mundo. Muchas de estas herramientas requieren el despliegue de datos 3D generados por tomógrafos y la interacción del usuario con los mismos para facilitar determinar las causas de las patologías en los pacientes.

Es por ello que es importante el desarrollo una arquitectura para el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos; el cual se va a lograr cumplir mediante el análisis de la captura de datos generados por los tomógrafos y del proceso de despliegue tridimensional en dispositivos móviles, el diseño de la arquitectura básica para cada uno de los componentes requeridos para el funcionamiento del prototipo, el desarrollo de un prototipo básico para el despliegue tridimensional de archivos STL (Stereolithography) sobre la arquitectura propuesta, efectuando las pruebas de rendimiento sobre el prototipo desarrollado para validar los tiempos de respuesta bajo las características básicas mínimas de red y del dispositivo móvil para el despliegue del modelo.

Para logra los objetivos trazados en cuanto al prototipo, se hizo uso de herramientas de cómputo para la elaboración de aplicaciones en Android, particularmente el entorno de desarrollo para aplicaciones Android Studio 1.0 que es gratuito bajo la licencia Apache 2.0 en un entorno Windows 7 Home Basic el cual utiliza como lenguaje de programación orientado a objetos Java. Adicionalmente se tomó la aplicación denominada *STL Viewer for Android* como librería para la visualización de objetos 3D que utiliza OpenGL como estándar de API gráfica multiplataforma para el procesamiento tridimensional. Además, para realizar la carga de datos y modelos 3D que son consultados por el prototipo móvil, se encuentra un sistema web desarrollado en PHP 5.6.3 bajo un servidor web Apache 2.4.10 que sirve como plataforma para realizar las conexiones bidireccionales con el cliente mediante el protocolo HTTP, que a su vez se conecta a la base de datos centralizada Mysql versión 5.6.21.

Esto beneficiará a la comunidad científica que desee profundizar y/o continuar con el diseño y especificación de la arquitectura propuesta, u otros afines tales como: planteamiento y características del despliegue 3D de imágenes médicas, planteamiento y programación en dispositivos móviles, entre otros. Así como también beneficiará a las instituciones de salud pública y privada obtener beneficios en forma de optimización de procesos, poseer una

herramienta de fácil acceso, de bajo costo y de una manera más cómoda sin los límites institucionales que restringen a los métodos tradicionales el intercambio de imágenes médicas para agilizar la colaboración con los pacientes durante el proceso de decisión en la atención médica.

En cuanto a metodología cabe exponer que el tipo de investigación del presente proyecto es aplicativa bajo el método en espiral de Boehm ya que consiste en proveer de la implementación de la propuesta tecnológica la cual será funcional y producto derivado de los conocimientos teóricos. Asimismo presenta un diseño bibliográfico, debido a que los datos que se usarán para el desarrollo de la arquitectura para el despliegue tridimensional en dispositivos móviles serán obtenidos de fuentes secundarias. La técnica de recolección de tipo revisión documental y entrevista no estructurada con los expertos en el área médica, ya que los datos e información requeridos para la consecución de los objetivos se obtendrán de planteamientos y estudios plasmados en documentos impresos y vía web.

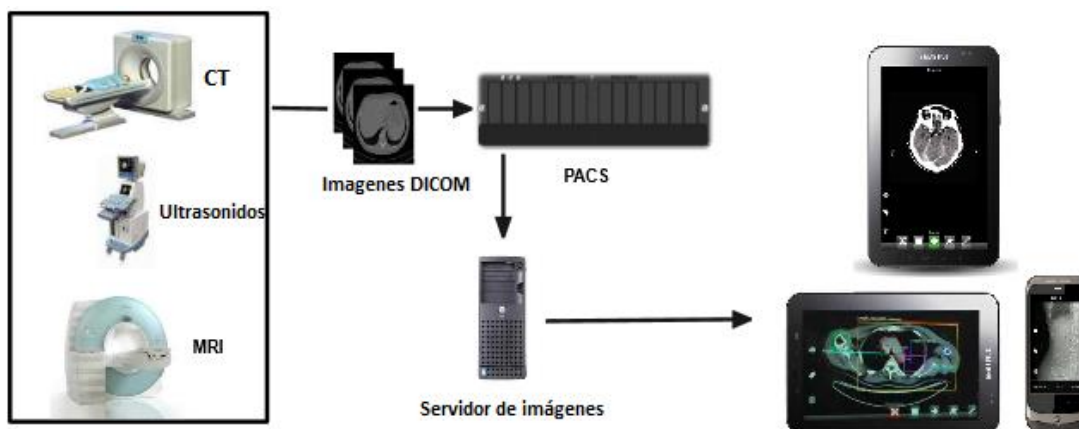
El presente proyecto de grado consta del planteamiento del problema, los antecedentes o trabajos previos vinculados con el problema, relevancia o justificación de la investigación, metodología que se llevó a cabo durante el trabajo de investigación, los resultados esperados a alcanzar mediante el trabajo de investigación y los resultados del proyecto.

## **1.2 Antecedentes**

Para el desarrollo de la presente investigación se toman como referencia trabajos elaborados por diferentes autores en el ámbito nacional e internacional que sirve como apoyo referencial al presente estudio el cual lleva por título Arquitectura para el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos. Como se mencionó, en el ámbito internacional se han publicado diversos artículos relacionados con el tema de estudio, entre los que se pueden mencionar:

Fermi, Supramaniam, Kuo, Ammar, Anil y Rajeswari (2012) presentan “**An Android-based Mobile Medical Image Viewer and Collaborative**” el cual es un sistema que permite a

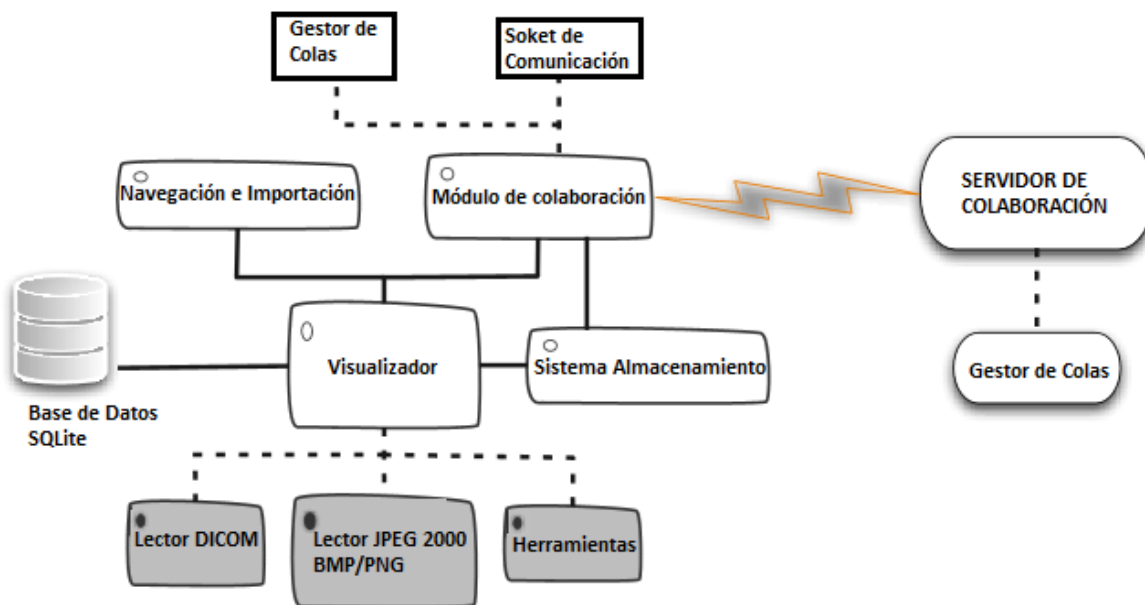
múltiples expertos médicos ver, analizar y discutir las regiones de interés en imágenes médicas de forma remota a través de dispositivos móviles. La investigación está centrada en la creación de mecanismos efectivos para acceder de forma segura a expertos para dar opiniones sobre las imágenes médicas en tiempo real a través de la web con tecnologías para las discusiones en línea. El trabajo propone un prototipo de colaboración en línea para móviles usando el sistema operativo Android 2.3. La Figura 1 muestra el flujo de trabajo que ellos proponen, comenzando con imágenes médicas que pueden ser obtenidas de diferentes fuentes, como la tomografía computarizada, ultrasonido, resonancia magnética, entre otros. Estas imágenes se encuentran en formato DICOM (Digital Imaging and Communication in Medicine) y se envían al Sistema de Archivo y Comunicación de Imagen (PACS). Los usuarios utilizan estaciones de trabajo DICOM con capacidad de obtener las imágenes de los PACS para su posterior visualización o diagnóstico. Para las pruebas se utilizó un servidor de colaboración para lograr tomar las imágenes desde los PACS y poderlas visualizar en el sistema móvil.



**Figura 1:** Experimentación del flujo de trabajo de proyecto (Tomado de Fermi, Supramaniam, Kuo, Ammar, Anil y Rajeswari, 2012)

Entrando un poco más en detalle, la arquitectura del visor de imágenes médicas móviles basada en Android 2.3 tiene cuatro componentes principales: el Visualizador, el sistema de almacenamiento, La navegación e importación, y el Módulo Colaborativo. Para la base de datos se utilizó SQLite para almacenar datos con un servidor de colaboración ligero. El componente Visor es básicamente el componente principal del sistema, vincula los otros componentes y

gestiona el flujo de datos entre el usuario y el sistema. El diagrama 1 representa la arquitectura utilizada:



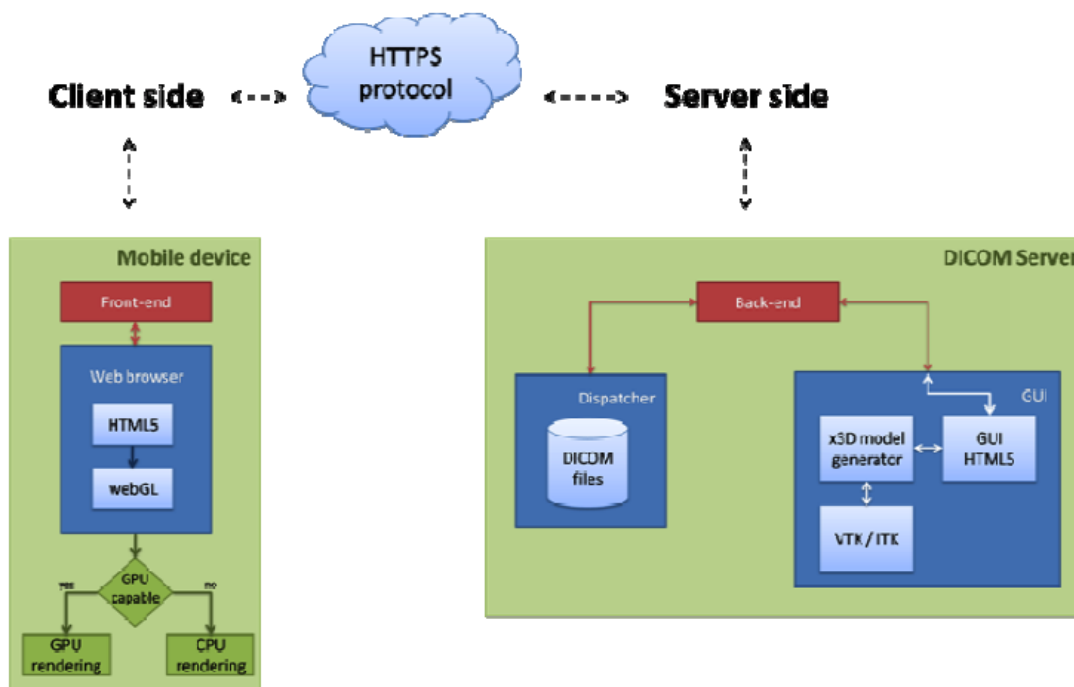
**Figura 2:** Diagrama de la arquitectura del sistema propuesto por Fermi y otros (Fermi, Supramaniam, Kuo, Ammar, Anil y Rajeswari, 2012)

Fermi, Supramaniam, Kuo, Ammar, Anil y Rajeswari concluyen que desarrollar un visor de imágenes médicas móviles presenta desafíos importantes ya que ayuda a facilitar la consulta y diagnóstico de los especialistas médicos pudiendo consultar la información en cualquier momento. Estos expertos siempre pueden permanecer conectados dondequiera que se encuentren a través de la tecnología móvil (especialmente durante emergencias o momentos críticos). Esta característica de colaboración, ayuda en la prestación de un mejor diagnóstico y planificación quirúrgica y seguimiento a los procedimientos. Sin embargo, esta arquitectura no cuenta con un despliegue tridimensional de las imágenes médicas en los dispositivos móviles ni se integra con el sistema de información y/o sistema radiólogo del centro hospitalario.

Ojog y Arias (2012) presentan “**m3DICOM: A Platform for mobile DICOM Visualization Based on X3D**” el cual proponen una plataforma adecuada para Visualización DICOM en móviles, superando algunas de las limitaciones actuales de los visores DICOM

basadas en la Web mediante la integración de modelado en 3D en la nube, a través de la generación un archivo X3D para la visualización. Ellos indican que los visores DICOM basado en la Web en tiempo real son deseables, pero la limitación del ancho de banda en altos volumen de datos es una barrera difícil de pasar. Sin embargo, con las nuevas tecnologías de las tabletas “Tablet” o teléfonos inteligentes, y la nueva generación de navegadores basados en WebGL con GPU para la aceleración gráfica, es posible explorar otras alternativas; es por ello que el archivo DICOM se alojó del lado del servidor y entrega un modelo dinámico X3D al dispositivo móvil para la visualización e interacción con el usuario. El procesamiento se lleva a cabo en la nube, pero la Representación 3D final se realiza en el dispositivo móvil.

La plataforma propuesta por Ojog y Arias se basa en una arquitectura cliente-servidor que permite a los médicos o radiólogos visualizar, analizar e interactuar con la información del paciente almacenada en un repositorio DICOM web, con el uso de un dispositivo móvil y un navegador WebGL habilitado. La arquitectura general se representa en la siguiente Figura 3, el cual el servidor DICOM contiene una base de datos de pacientes, y las aplicaciones para procesar en la nube los archivos para extraer una Modelo X3D partir de los datos. El modelo 3D es creado por un proceso de segmentación y cortes de los datos DICOM. Seguidamente un modelo complejo se ensambla y convierte a formato HTML5 para ser visualizados en el lado del cliente con un navegador compatible con WebGL. El dispositivo cliente podría integrar la aceleración de hardware de gráficos para la representación 3D local (Ojog y Arias, 2012)



**Figura 3:** Arquitectura para el visor DICOM en X3D (Tomado de Ojog y Arias, 2012)

La implementación de la plataforma que se llevó a cabo en el lado del servidor, los archivos DICOM se almacenan en un sistema de archivos básico, la aplicación se comunica con el cliente para recibir los parámetros de visualización y realiza la segmentación DICOM, para entregar del modelo X3D. El procesamiento se lleva a cabo utilizando la bibliotecas del Kit de herramientas de visualización (VTK, en inglés, Visualization ToolKit), ITK (Insight Toolkit) y algoritmos propietarios utilizados para segmentar el modelo 3D a partir de los datos de DICOM. Por otra parte, en el dispositivo móvil, el usuario puede acceder a la página web a través de una interfaz de navegador compatible con HTML5. Para probar los resultados se utilizó una tabla modelo Viewsonic con Android 2.2 y el navegador Fennec 4.0b6pre. El problema presentado en esta arquitectura es la integración entre un sistema que mantenga la información básica de los pacientes y el modelo 3D de su respectivo estudio, ya que el despliegue está ligado al navegador instalado en el dispositivo móvil y solo se enfoca en la visualización de dicho modelo tridimensional.

Goretzki (2014) presenta su investigación llamada “**3D Reconstruction of Medical Images Using Java3D**” en donde analiza las técnicas renderización comunes orientadas en el



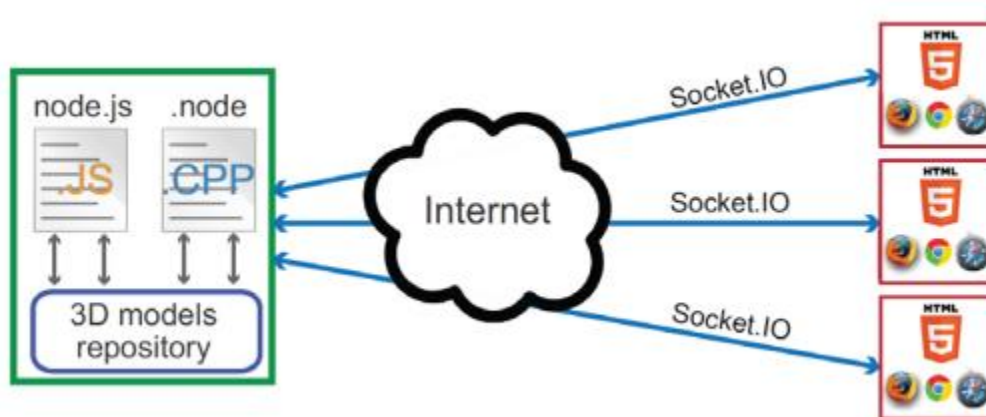
uso para el sector médico. Varias técnicas se describen tanto para la superficie y renderización volumétrica. El problema principal se encuentra en mostrar datos de imágenes volumétricas en modelos 3D tan rápido y preciso como sea posible. La API basada en JAVA 3D se utiliza para implementar el proyecto, que es capaz de realizar la renderización de volúmenes. Este programa muestra una pila de imágenes como un modelo 3D para que el usuario sea capaz de interactuar con él, pudiendo girar, mover o escalar en el interior del espacio 3D. Otra posibilidad importante es dividirlo en pedazos para obtener una visión interior del volumen. El resultado de la investigación responde a la pregunta sobre si Java 3D es lo suficientemente potente para llevar a cabo la renderización volumétrica en poco tiempo con un uso de memoria aceptable y solo se basa en determinar el uso de diferentes técnicas para dicha renderización volumétrica usando JAVA 3D.

Adicionalmente se pueden encontrar algunos desarrollos en el ámbito nacional, en el que se cuenta la publicación llamada “**Web visualization of 3D medical data with open source software**” propuesto por Esmir Ramírez y Ernesto Coto (2012) del Centro de Computación Gráfica de la Escuela de Computación de la Facultad de Ciencias de la Universidad Central de Venezuela. Allí proponen los websockets para proveer una base para desarrollar aplicaciones para la web con despliegue tridimensional ejecutada sobre cualquier plataforma. Este trabajo propone cuatro arquitecturas para el desarrollo de aplicaciones web para la visualización de datos 3D, basadas en la tecnología actual de código abierto de websockets describiendo la implementación detallada de una de las arquitecturas y presentando pruebas y resultados para mallados 3D y volúmenes. El sistema puede ser utilizado para desplegar volúmenes y mallados médicos en tiempo real sobre la web, contribuyendo potencialmente al área de Telemedicina.

Para la implementación del sistema de visualización web 3D propuesto por Ramirez y Coto, se eligió la Arquitectura descrita por la Figura 4, cuyo enfoque consiste en la creación de un servidor HTTP para cargar un modelo y renderizarlo fuera de él. Después de eso, esta imagen se envía a un navegador en el que el usuario puede interactuar, cada vez que el usuario realiza una acción (rotación, zoom, entre otros) un conjunto de comandos y parámetros se envía al servidor, el cual se aplica la operación y devuelve una nueva imagen 2D. Todos los datos 3D son almacenados en el servidor, además existe una extensión desarrollada llamada Node.js para

elaborar el renderizado. En el lado del cliente, el navegador recibe la imagen y la muestra utilizando un componente HTML5. La comunicación entre el servidor y el cliente se hace usando Socket.IO, la implementación de WebSocket Node.js. El protocolo de comunicación se basa en TCP en vez de en HTTP.

Entre las desventajas que presenta la arquitectura dada por Ramírez y Coto (2012) es que la investigación se basa en la visualización 3D de imágenes médicas sobre la Web, más no en los dispositivos móviles ni en la forma de integrar los elementos que comprende la infraestructura DICOM y sistemas de información radiográficos presentes en los centros asistenciales médicos.



**Figura 4:** Arquitectura para el despliegue 3D (Tomado de Ramírez y Coto, 2012)

La siguiente tabla muestra una comparación de las principales características presentadas entre los diferentes antecedentes anteriormente mencionados por cada uno de los autores:

	Fermi y Otros (2012)	Ojog y Arias (2012)	Goretzki (2014)	Ramírez y Coto (2012)
<b>Repositorio de Modelos</b>	Si	Si	Si	Si
<b>Despliegue 3D</b>	No	Si	Si	Si
<b>Despliegue Móvil</b>	Si	No	No	No
<b>Plataforma Libre</b>	No	No	No	Si

<b>Sistema Web para Carga de modelos 3D y pacientes</b>	Si	No	No	No
<b>HTML5</b>	No	Si	No	Si
<b>Sistema de Colaboración</b>	Si	No	No	No
<b>Plataforma Android</b>	Si	No	No	No
<b>Arquitectura Cliente-Servidor</b>	No	Si	No	Si
<b>Almacenamiento Base de Datos</b>	Si	No	No	No
<b>Interconexión de dispositivos en la red</b>	Si	Si	No	Si

**Tabla 1.** *Tabla comparativa de arquitecturas antecedentes*

### **1.3 Planteamiento del problema.**

Es necesario que las organizaciones de hoy en día afronten sus problemas de una manera eficaz y eficiente, que les permita alcanzar sus objetivos de corto, mediano y largo plazo (Whitten, 1997). A tal efecto es indispensable el uso de las tecnologías de la información (TI), que según González (Centro Nacional de Tecnología de Información, 2014), son “El conjunto de procesos y productos derivados de las nuevas herramientas (hardware y software), soportes de la información y canales de comunicación relacionados con el almacenamiento, procesamiento y transmisión digitalizados de la información”.

Especialmente las tecnologías móviles tienen bastante tiempo entre las personas permitiendo simplificar muchas de las actividades cotidianas y facilitando trabajos, estudios, comunicación, localización, entre otros, con sus múltiples aplicaciones disponibles que se han ido incrementando con el tiempo, representando adicionalmente beneficios financieros y operacionales para las empresas, ya que las tendencias muestran que el costo de dispositivos móviles y comunicación de datos continuaran bajando en los próximos años, lo cual indica que

cada vez más las empresas pasarán a utilizar sus dispositivos móviles para enviar datos referentes a sus negocios.

Según la Comunidad Virtual de Anatomía Patológica (2014), existe una competencia cada vez más intensa en el cuidado de la salud como en la mayoría de las industrias. Es importante reducir los costos y necesario hacer más con menos recursos, lo que implica la búsqueda activa de soluciones que permitan resolver eficazmente atender a más pacientes con el mismo, o incluso menos personal; encontrar modos de atraer a nuevos clientes con el caso de instituciones con fines de lucro y desarrollar o implementar tecnologías que permitan la reducción de costes globales y aumento de la productividad.

Adicionalmente existen sistemas para la captura, gestión, tratamiento y almacenamiento de imágenes médicas resultantes de distintos estudios como los PACS (Picture Archiving and Communications System). Con el surgimiento de la tomografía computarizada, se ha incrementado la cantidad de estudios así como la cantidad de equipos médicos capaces de adquirir, almacenar y visualizar imágenes médicas, con el objetivo de permitir el funcionamiento de un servicio de imagen sin necesidad de documentos en papel, integrando las imágenes y la información clínica en un sistema central de gestión y archivo conectados en una red de comunicación (Franco, 2011).

Gracias a las altas velocidades de procesamiento que se tienen actualmente en los computadores de mano modernos, se comprende la importancia de obtener soluciones móviles que permitan desplegar de forma tridimensional datos generados por un tomógrafo para la mejora de procesos en instituciones de salud pública y privadas a través de una plataforma y herramienta de fácil acceso, de bajo costo, la flexibilidad de acceder a las imágenes en cualquier momento y de una manera más cómoda sin los límites institucionales que restringen a los métodos tradicionales el intercambio de imágenes médicas para agilizar la colaboración con los pacientes durante el proceso de decisión en la atención médica. Del mismo modo, necesitan proveerse de aplicaciones que añadan valor a sus áreas de trabajo y eviten el uso de tecnologías propietarias para su realización con la más alta calidad, excelente servicio y reduciendo el costo del proceso.

El sector médico cuenta con grandes avances en la investigación y desarrollo de tecnologías que permiten mantener el estado de salud en los pacientes, sin embargo, hoy en día aún existen fallecimientos o desmejora del estado crítico por falta de información en su traslado a un centro médico, aunado a la escases de las placas radiográficas donde se imprimen las tomografías o disco compacto donde se guardan en formato digital. Es de suma importancia determinar la sintomatología del paciente sin la necesidad de largos traslados por parte del médico para canalizar por grado de riesgo o enfermedad, a los pacientes que requieren de una atención médica inmediata contando con información previa y efectiva con la que se lograra detectar que su vida corre peligro.

En este orden de ideas se advierte la necesidad del desarrollo de una arquitectura para el despliegue tridimensional en dispositivos móviles de datos generados por un tomógrafo en una plataforma libre (en este caso Android 4.2.2) que proporcione una mayor flexibilidad a terceros sobre cuestiones de desarrollo y concesión de licencias, soportado por la mayoría de dispositivos móviles creados hoy en día, en comparación con otros enfoques cerrados como iOS de Apple o Windows Phone, el cual estaría disponible para el personal experto de las clínicas y hospitales que cuente con un teléfono celular o tabla, logrando anticipar, atender y monitorear el manejo de enfermedades o traumatologías presentadas, y a su vez permitiendo manipular los estudios, rotando, ampliando y navegando a través del cursor, permitiendo de esa forma analizar mejor los resultados del examen y disminuir los costes de la atención sanitaria para beneficiar a los pacientes, lo que se traduce en una reducción o eliminación de la película o discos compactos con disponibilidad de los datos para el especialista o médico de familia, que será ahora inmediata.

## **1.4 Objetivos.**

### **1.4.1 Objetivo General.**

Desarrollar una arquitectura para una plataforma computacional que permita el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos.

### **1.4.2 Objetivos Específicos.**

- Analizar la captura de datos generados por los tomógrafos.
- Analizar el proceso de despliegue tridimensional en dispositivos móviles.
- Diseñar la arquitectura básica para cada uno de los componentes requeridos para el funcionamiento del prototipo.
- Desarrollar un prototipo básico para el despliegue tridimensional de archivos STL (Stereolithography) sobre la arquitectura propuesta.
- Realizar las pruebas de rendimiento sobre el prototipo desarrollado.
- Establecer las características básicas mínimas de red y del dispositivo móvil para el despliegue del prototipo.

### **1.5 Alcance.**

El alcance del proyecto es presentar una propuesta básica para una arquitectura de una plataforma programada que suministre el soporte para el despliegue tridimensional de imágenes generadas por tomógrafos en formato DICOM y el desarrollo de un prototipo que cumpla los requisitos mínimos de los elementos compuestos por dicha arquitectura, bajo plataforma Android 4.2.2 para dispositivos móviles sobre el formato STL (Stereolithography), siendo la plataforma propuesta escalable, en código libre, ligera y abierta al uso de otras tecnología para el despliegue de los modelos de estudio tomando en cuenta cada etapa del proceso de captura de datos hasta el diagnóstico de los resultados por parte del especialista médico.

### **1.6 Justificación.**

Hoy en día, las imágenes médicas producidas por los tomógrafos u otros dispositivos como la resonancia magnética o la tomografía por emisión de positrones son de gran ayuda para diagnosticar diversos trastornos patológicos que afectan a los seres humanos. En la actualidad, existen numerosos sistemas que permiten a los profesionales de la medicina acceder a dichas imágenes ubicadas en servidores especializados a través de estaciones de trabajos o dispositivos móviles como los teléfonos inteligentes o tablas desde cualquier lugar con conexión a internet.

Existen numerosas soluciones gratuitas y comerciales que permiten el despliegue tridimensional de datos generados por los tomógrafos en estaciones de trabajos (como las

portátiles y computadores de escritorio), sin embargo, no poseen la movilidad, comodidad y versatilidad que cuentan los dispositivos móviles, que permitirían al equipo médico calificado efectuar diagnósticos a sus pacientes desde cualquier lugar con acceso a la red de una manera mucho más cómoda y práctica.

Igualmente, existen diversas soluciones en el mercado que ofrecen servicios móviles, entre los que se pueden mencionar SonicDicom de la empresa llamada JIUN (JIUN Corporation, 2016) y mRay de desarrollado por mbits (mbits, 2016) pero que son muy costosos y que muchos centros de salud no se encuentran en la capacidad de pagar, por ello es necesario el desarrollo de una alternativa de más fácil acceso para visualizar de manera tridimensional, en dispositivos móviles, los datos generados por un tomógrafo sin los límites geográficos o institucionales que restringen los métodos tradicionales para el intercambio de imágenes médicas agilizando la colaboración con los pacientes durante el proceso de decisión en la atención médica.

El resultado de la implementación de este tipo de soluciones, se traduce en una reducción o eliminación de la película o discos compactos y sus costes asociados, así como mayor disponibilidad de los datos para el especialista o médico de familia, que será ahora inmediata. Como es actualmente, los radiólogos desde su estación de diagnóstico realizan la lectura de todos los estudios de su lista de trabajo utilizando monitores de alta resolución y software específico, que les permite la visualización de mayores detalles. El resto de los médicos, podrán acceder al estudio y su información asociada desde su oficina o casa utilizando un dispositivo móvil. La institución se beneficia en aumento de la productividad, así como de una mayor satisfacción de sus clientes, gracias a las mejoras en calidad y rapidez del proceso asistencial completo (Comunidad Virtual de Anatomía Patológica, 2014).

Cabe destacar que el proyecto es factible de realizar puesto que se cuenta con la infraestructura de hardware (celular, periféricos, conexiones de red), así como las herramientas de software (compiladores, servidores y entornos de desarrollo de aplicaciones) y el personal académico asesor en la materia que ayudaran a desarrollar las habilidades requeridas sobre el tema. Además metodológicamente, por ser un área afín de la programación, se cuenta con las bases de conocimiento necesarias para emprender la programación requerida.

## **1.7 Organización del Documento.**

El documento está estructurado en seis capítulos. En el Capítulo I: se detalla la introducción, antecedentes, planteamiento del problema, justificación, objetivos y alcance del proyecto. El Capítulo II: contiene el marco teórico la cual sirve como base para el desarrollo de la arquitectura propuesta; en cuanto, al Capítulo III: se refiere a la metodología empleada en el proyecto de tesis; Capítulo IV: se presenta el diseño de la arquitectura; Capítulo V: se describe la implementación de la arquitectura; Capítulo VI las pruebas y Capítulo VII: Conclusiones y recomendaciones.



## **CAPÍTULO II**

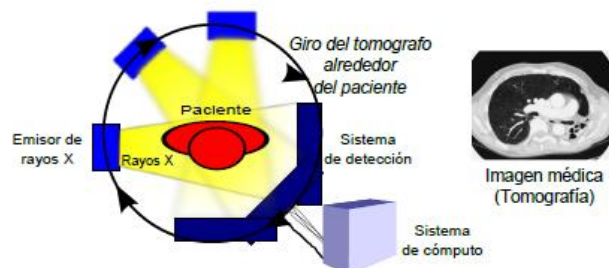
### **MARCO TEÓRICO**

La arquitectura propuesta para el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos consta de una serie de elementos conceptuales que son importantes conocer para fundamentar la investigación, la interrelación entre cada uno de ellos y entender los formatos y estructuras manejados.

#### **2.1 Elementos conceptuales**

##### **2.1.1 Tomografía.**

Para el desarrollo del trabajo y de la arquitectura que soporta el prototipo en los dispositivos móviles, es necesario desde un comienzo realizar el estudio sobre el área de interés afectada del paciente para que los especialistas puedan tomar una decisión. Para ello es requerido que dichos pacientes se sometan a un procedimiento de diagnóstico por medio de una tomografía que permite el procesamiento de imágenes por secciones a través de un aparato llamado tomógrafo. Este método es usado en medicina, arqueología, biología, geofísica, oceanografía, ciencia de los materiales y otras ciencias. En la mayoría de los casos se basa en un procedimiento matemático llamado reconstrucción tomográfica que permite obtener imágenes transversales de pacientes. Un segmento plano del cuerpo se hace irradiar en varias direcciones por haces de rayos X, y a partir de la detección de los haces transmitidos a través de la sección, se reconstruye una imagen plana del segmento, tal como si este hubiese sido físicamente removido del cuerpo irradiado en una vista perpendicular (Wikipedia, 2014), como se ilustra en la Figura 5:

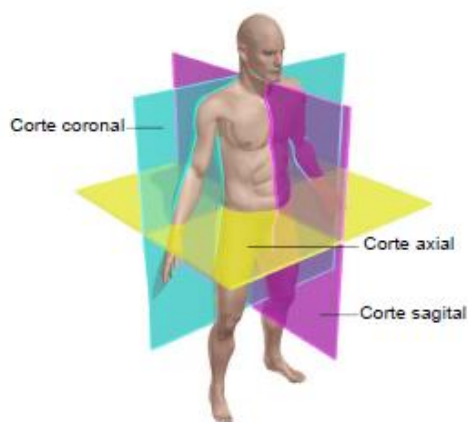


**Figura 5:** *Diagrama de tomografía (Tomado de Franco, 2011)*

Según el tipo de estudio que se desee realizar se puede generar diferentes tipos de cortes, entre ellos se tienen:

### 2.1.2 Tipos de cortes en la tomografía computarizada.

La tomografía computarizada es una técnica de diagnóstico por imagen que permite la visualización de cortes del organismo a partir de múltiples determinaciones de absorción de rayos X. La formación de la imagen se limita principalmente a cortes transversales (axiales) perpendiculares al eje longitudinal del cuerpo; aunque la reconstrucción final puede obtenerse en cualquiera de los tres planos (o cortes) anatómicos; axial, coronal y sagital, como se muestra en la Figura 6:



**Figura 6:** *Tipos de corte de las tomografías (Tomado de Franco, 2011)*

### **2.1.3 Tipos de Tomógrafos**

Según la calidad, rapidez y necesidad del estudio que se desee para determinar la patología presente en el cuerpo del paciente, existen distintos tipos de tomógrafos que se pueden encontrar en el mercado, entre ellos se pueden mencionar:

#### ***2.1.3.1 Tomógrafos helicoidales.***

Rotan continuamente en una dirección mientras el paciente se mueve por el túnel del escáner y permite obtener cortes más delgados del área analizada, lo que mejora el diagnóstico médico (Rademacher, 2014).

#### ***2.1.3.2 Tomógrafos multicorte.***

Permite a los médicos capturar imágenes de varios cortes durante la rotación del tubo de rayos X del tomógrafo, el cual acelera el tiempo necesario para completar un análisis por tomografía comparado con el tomógrafo helicoidal. Las imágenes proporcionadas por los tomógrafos multicorte son de cortes más delgados que aquellas obtenidas por el tomógrafo helicoidal (Rademacher 2014).

#### ***2.1.3.3 Tomografía computarizada por haz de electrones.***

Un rayo de electrones se crea mediante un generador el cual elimina la necesidad de partes mecánicas móviles y proporciona los tiempos de escaneo más rápidos. Los tomógrafos con rayo de electrón se utilizaron por primera vez para analizar órganos móviles, como el corazón y los pulmones. La capacidad rápida del escáner de este tomógrafo permite capturar imágenes fijas aunque el órgano del paciente se esté moviendo.

### **2.1.4 Sistema de Archivado y Transmisión de Imágenes (PACS)**

Los estudios generados por los tomógrafos requieren de un sistema que posibilite a los médicos especialistas la captura, gestión, tratamiento y almacenamiento de las imágenes médicas

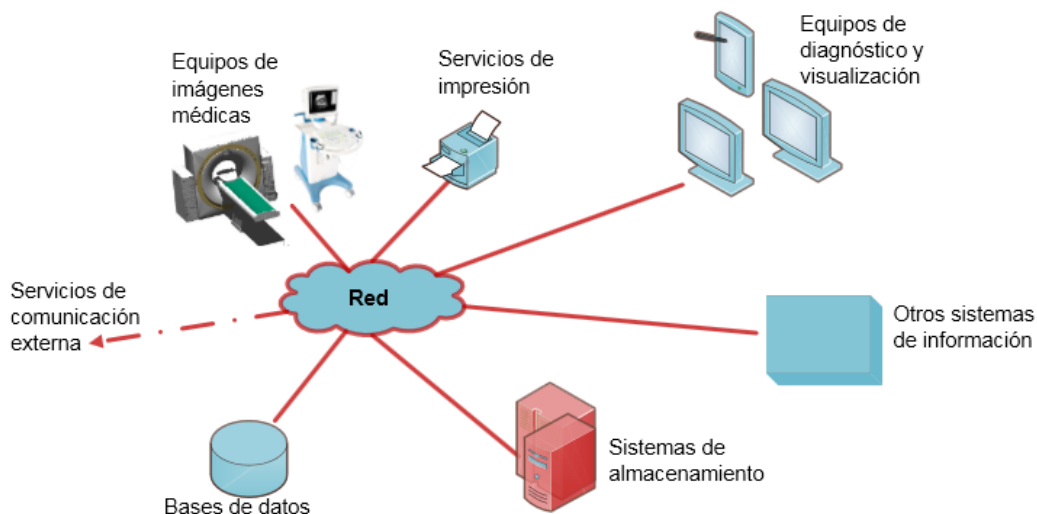
resultantes para su posterior análisis de los resultados obtenidos, para ello, surgen los PACS (Picture Archiving and Communications System).

Con el nacimiento de la tomografía computarizada, se ha incrementado la cantidad de estudios imagenológicos así como la cantidad de equipos médicos capaces de adquirir, almacenar y visualizar imágenes médicas. Al conjunto de equipos informáticos dedicados a la adquisición, almacenamiento, procesado y comunicación de imágenes médicas digitales junto a su información asociada se denominan PACS, su objetivo es permitir el funcionamiento de un servicio de imagen sin necesidad de imágenes en película ni documentos en papel, integrando las imágenes y la información clínica en un sistema central de gestión y archivo conectados en una red de comunicación.

Los Sistemas PACS ofrecen una alternativa en el manejo de imágenes digitales a través de dispositivos conectados en red para estudios de:

- Rayos X Tomografía
- Ecocardiografía
- Angiografía
- Ultrasonido
- Medicina Nuclear

Los sistemas PACS, utilizan varios componentes de hardware y software con componentes básicos que comprenden equipos de composición de imágenes, red de comunicación, base de datos, estaciones de diagnóstico y visualización, y sistemas de almacenamiento, como se muestra en la Figura 7:



**Figura 7:** Diagramas de sistemas PACS (Tomado de Franco, 2011)

### 2.1.5 Estándares para imágenes médicas

Según Moyano y Quiles (1997), dada a las grandes cantidades de datos manipulados por los hospitales e instituciones de atención sanitaria, es necesario que cuenten con un estándar en común para comunicarse entre las distintas tecnologías y dispositivos de despliegue. Dichos estándares incluyen los formatos de las imágenes, los algoritmos de compresión y las comunicaciones y estándares de red. Entre los más importantes se pueden mencionar los siguientes:

- Estándares específicos para la comunicación de imágenes médicas como DICOM (Digital Imaging and Communication in Medicine) y MEDICOM (Medical Image Communication).
- Otros estándares de imágenes médicas que han sido implementados con poco éxito, como IS&G e INTERLIFE.
- Estándares generales sobre imágenes, como IPI (Image Processing and Interchange).
- Estándares de compresión de imágenes generales, como por ejemplo JPEG y JBIG.
- Estándares de comunicación generales para la comunicación en internet.

### 2.1.5.1 Estándar DICOM

Para la Facultad de Informática de la Coruña (Sin Fecha) el estándar DICOM es utilizado para el intercambio de pruebas médicas, para el manejo, visualización, almacenamiento, impresión y transmisión de datos. DICOM permite la integración de escáneres, servidores, estaciones de trabajo, impresoras y hardware de red de múltiples proveedores dentro de un sistema de almacenamiento y comunicación de imágenes.

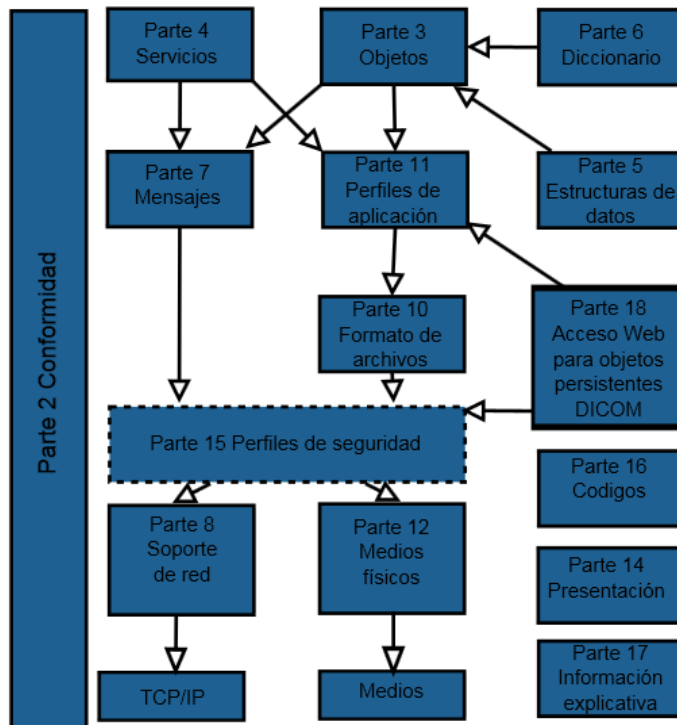
Corresponde al estándar más importante y extendido de la industria para la transferencia de imágenes e información médica entre dispositivos electrónicos. Permite integrar al usuario un equipo entre los distintos dispositivos de diferentes fabricantes para soportar el amplio rango de modalidades de imágenes médicas, evitando los problemas de compatibilidad que se puedan presentar entre los fabricantes y protocolos.

Para Moyano y Quiles (1997), éste estándar fue desarrollado por la ARC (The American College of Radiology) responsable de otorgar la dirección técnica y asesoramiento médico y NEMA (The National Electrical Manufacturers Association), encargada de actuar como entidad de publicación proporcionando asesoramiento legal para evitar conflicto de intereses o violaciones en el ámbito de trabajo. Ambas asociaciones se unieron para desarrollar DICOM con los siguientes objetivos:

- Establecer un estándar para la comunicación en entorno de red.
- Establecimiento de los requerimientos mínimos para otorgar una base legal con el estándar a todos fabricantes que lo solicitaran.
- Permitir inter-operatividad entre distintos equipos de los fabricantes mediante la introducción de objetos de información que corresponde a las imágenes, estudios, informes, entre otros.

Según la Facultad de Informática de la Coruña (Sin Fecha), en la actualidad, el estándar DICOM se encuentra especificado en 18 partes, como se muestra en la Figura 8:

- Parte 1: Introducción
- Parte 2: Conformidad
- Parte 3: Definiciones de objetos de información
- Parte 4: Especificaciones de clases de servicio
- Parte 5: Estructuras de datos y codificación
- Parte 6: Diccionario de datos
- Parte 7: Intercambio de mensajes
- Parte 8: Red de apoyo a la comunicación para el intercambio de mensajes
- Parte 9: Retirada (soporte de la comunicación punto-a-punto para el intercambio de mensajes)
- Parte 10: Soportes de almacenamiento y formato de archivo para los medios de intercambio
- Parte 11: Perles de aplicación de medios de almacenamiento
- Parte 12: Formatos de los medios de comunicación y medios físicos para los medios de intercambio
- Parte 13: Retirada (soporte de comunicación punto-a-punto para gestión de la impresión)
- Parte 14: Función de escala de grises para visualización en pantalla
- Parte 15: Seguridad gestión de perles
- Parte 16: Cartografía de los recursos de contenido
- Parte 17: Información explicativa
- Parte 18: Acceso Web para objetos persistentes DICOM (WADO)



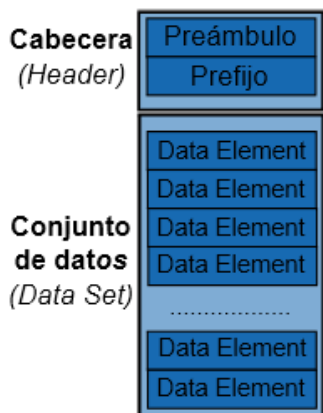
**Figura 8:** *Estándar DICOM (Tomado de Franco, 2011)*

### 2.1.5.2 Formato de una archivo DICOM

Martínez (2006) indica que la estructura de un archivo DICOM consiste en dos partes:

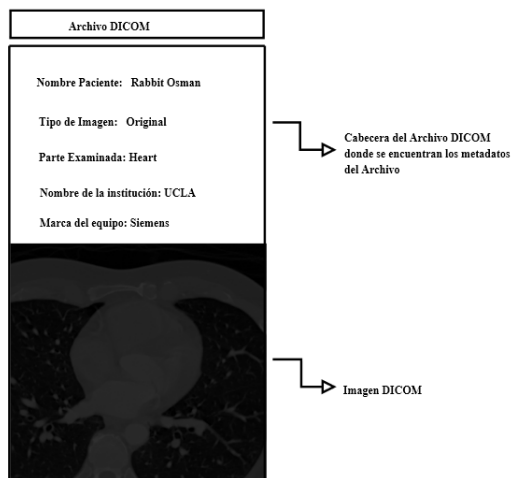
- Una cabecera con multitud de campos estandarizados que especifican datos administrativos (datos del paciente, hospital donde se realizó, edad, identificador, tamaño de la imagen, dimensiones, modalidad, equipo de captura, entre otros), datos sobre el estudio y la sintaxis de transferencia UID (identificador único) que especifica la codificación y la compresión del conjunto de datos que le sigue.
- Un conjunto de datos (data set) de DICOM, que contiene la imagen o las imágenes especificadas que pueden estar comprimidas con distintos estándares.





**Figura 9:** Estructura de Archivo DICOM (Tomado de Martínez, 2006)

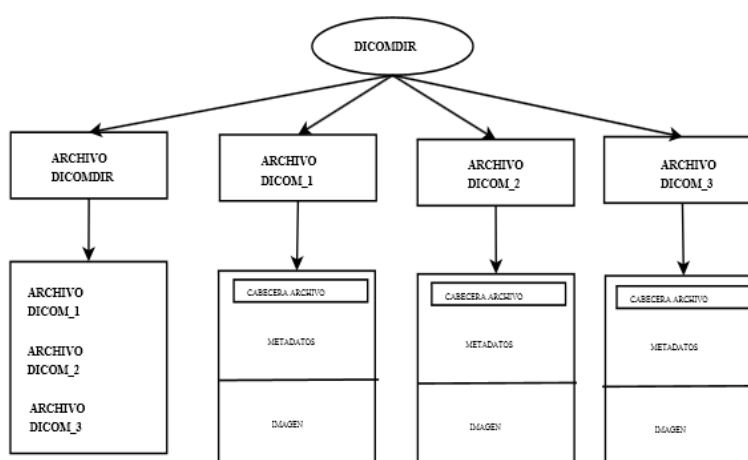
En los archivos DICOM se comprimen los metadatos y la imagen, con lo que se permite un menor tamaño y menor riesgo de pérdida de la información. A diferencia de la manera convencional (imágenes impresas en placas), los archivos DICOM son de gran ventaja pues permiten tener un criterio más amplio en el estudio de casos, basándose en imágenes tomadas anteriormente al mismo paciente. También permite desplegar en la pantalla varias imágenes de diferentes estudios para ser observadas al mismo tiempo, ofreciendo un control más detallado y preciso. (Martínez, 2006) La Figura 10 muestra un ejemplo de la estructura de un archivo DICOM



**Figura 10:** Estructura archivo DICOM (Tomado de Martínez, 2006)

### 2.1.5.3 Estructura DICOMDIR

DICOMDIR (en inglés, Directorio DICOM) es un directorio dentro de la arquitectura del proyecto de tesis que contiene la información del conjunto de imágenes, el cual señala el grupo de archivos que están relacionados a través de los atributos de cada uno, facilitando el acceso a la información de las imágenes contenidas dentro de este. En la Figura 11 se ilustra la organización del formato DICOMDIR y de los archivos contenidos, se muestran archivos DICOM junto con su estructura básica de metadatos e imagen, que permite organizar los archivos como un árbol permitiendo un rápido acceso a la información.



**Figura 11:** Estructura Archivos DICOM (Tomado de Martínez, 2016)

Para la creación del modelo 3D que se genera (en este caso) sobre el estándar DICOM sobre el cual se hace el diagnóstico de los pacientes en el dispositivo móvil, se utiliza el formato de archivos STL, que se explica a continuación:

## 2.2 Formato de archivos STL

Para Burns (1989) STL (cuyo significado proviene del inglés "STereo Lithography") es un formato de archivo informático que describe la geometría de objetos tridimensionales en base a triángulos, sin información de color, texturas, capas, tipos de trazo, superficie o propiedades físicas que sí incluyen otros formatos CAD. Es una representación triangular de una geometría de la superficie de 3 dimensiones. La superficie descompone lógicamente en una serie de triángulos pequeños (facetas). Cada cara o faceta es descrita por una dirección perpendicular y

tres puntos que representan los vértices (esquinas) del triángulo. Estos datos son utilizados por un algoritmo de corte para determinar las secciones transversales de la forma 3-dimensional. Existen dos representaciones de este tipo de archivos, ASCII (acrónimo inglés de American Standard Code for Information Interchange, Código Estándar Estadounidense para el Intercambio de Información) y binaria, cuya última es más común dado a que el tamaño de la información es más compacto.

### 2.2.1 STL binario

Este es el formato utilizado para el despliegue 3D en el dispositivo móvil, ya que es más ligero permitiendo hacer mejor uso de los recursos de memoria y mayor rapidez para el movimiento, rotación y acercamiento del modelo. Usa el estándar IEEE para enteros y representación numérica de punto flotante. La sintaxis para un archivo STL binario es la siguiente:

```

UINT8[80] – Cabecera
UINT32 – Número de Triángulos

foreach triangle
REAL32[3] – Normal vector
REAL32[3] – Vértice 1
REAL32[3] – Vértice 2
REAL32[3] – Vértice 3
UINT16 – Atributo para la cantidad de bytes
End

```

Un archivo STL binario tiene una cabecera de 80 caracteres (que generalmente se ignora, pero nunca debe comenzar con la palabra "solid" ya que se va a asumir que se trata de un archivo STL ASCII). Después de la cabecera viene un entero largo sin signo de 4 bytes que indica el número de caras triangulares en el archivo. Luego se indica la información de cada triángulo en

un vector con la coordenada X, Y y Z para cada vértice. Al final va seguido por un entero sin signo que es para indicar la cantidad de bytes (Burns, 1989).

### 2.2.2 STL ASCII

El formato ASCII (acrónimo inglés de American Standard Code for Information Interchange, Código Estándar Estadounidense para el Intercambio de Información) está pensado principalmente para probar nuevas interfaces CAD. Su gran tamaño lo hace poco práctico para el uso general. La sintaxis de un archivo STL ASCII es el siguiente:

```

solid nombre
  facet normal  $n_i n_j n_k$ 
    outer loop
      vertex  $v1_x v1_y v1_z$ 
      vertex  $v2_x v2_y v2_z$ 
      vertex  $v3_x v3_y v3_z$ 
    endloop
  endfacet
endsolid nombre

```

Las palabras en negrita indican una palabra clave; éstos deben aparecer en minúscula. La notación "{...} +", significa que el contenido de las llaves se pueden repetir más de una vez. Los símbolos en cursiva son las variables que deben ser sustituidas por los valores especificados por el usuario. Los datos numéricos en las líneas de faceta normal y vértices son valores de precisión, por ejemplo, 1.23456E + 789 (Burns, 1989).

## 2.3 Servidor web

Un Servidor Web es un programa especialmente diseñado para transferir datos de hipertexto con todos sus elementos como textos, widgets, imágenes, entre otros utilizando el

protocolo http. El servidor web se encuentra alojado en un ordenador que cuenta con conexión a la red dentro de la arquitectura a la espera de que algún navegador (cliente web) le haga alguna petición, como por ejemplo, acceder a una página web y responde a la petición, enviando código HTML mediante una transferencia de datos en red. Dado a que es importante contar con software de código abierto en la arquitectura propuesta, se utiliza Apache versión 2.4.10 como servidor web ya que es multiplataforma, gratuito, robusto y que destaca por su seguridad y rendimiento bajo la licencia BSD (*Berkeley Software Distribution*, el cual permite modificar el código fuente siempre y cuando se reconozca su trabajo).

## **2.4 Repositorio de modelos 3D**

Un Repositorio es una infraestructura web capaz de brindar un conjunto de servicios a una comunidad, destinados a recopilar, gestionar, difundir y preservar contenidos a través de una colección organizada y accesible (De Giusti y Otros, 2012). La arquitectura propuesta contiene un repositorio centralizado de modelos 3D en formato .STL (Stereolithography) donde se almacenan en un directorio cada uno de los resultados de los estudios de los pacientes para su posterior despliegue en los dispositivos móviles.

## **2.5 Plataforma de interconexión**

Existen varios dispositivos que permiten interconectar elementos sobre redes inalámbricas, de forma que puedan interactuar entre sí. Entre ellos destacan los enrutadores, puntos de acceso inalámbrico, entre otros, para la emisión de la señal en la red y para la recepción se utilizan tarjetas para conectar a los computadores personales. Los puntos de acceso inalámbricos funcionan a modo de emisor remoto, es decir, en lugares donde la señal del enrutador no tenga suficiente radio, se colocan estos dispositivos, que reciben la señal bien por un cable de Par Trenzado sin Blindaje (UTP) que se lleve hasta él o bien que capture la señal débil y la amplifique. Los enrutadores se encargan de efectuar el reparto de la señal al igual que los concentradores y conmutadores, estos dispositivos son mucho más sencillos que los enrutadores, sin embargo, su rendimiento en la red local es inferior. (Basterretche, 2007)

## **2.6 Dispositivos Móviles**

Una gran cantidad de dispositivos electrónicos se clasifican actualmente como dispositivos móviles, desde teléfonos hasta tablets, pasando por dispositivos como lectores de identificación por radiofrecuencia (RFID). Entre las principales características que tienen los dispositivos móviles se puede mencionar que son aparatos pequeños; la mayoría de estos aparatos se pueden transportar en el bolsillo del propietario o en un pequeño bolso; tienen capacidad de procesamiento, tienen conexión permanente o intermitente a una red; tienen memoria (Memoria de Acceso Aleatorio, memorias flash, entre otros); normalmente se asocian al uso individual de una persona, tanto en posesión como en operación, la cual puede adaptarlos a su gusto y tienen una alta capacidad de interacción mediante la pantalla o el teclado. (Morillo, 2011)

Para Morillo (2011) en la mayoría de los casos, un dispositivo móvil puede definirse con cuatro características que lo diferencian de otros dispositivos que, aunque pudieran parecer similares, carecen de algunas de las características de los verdaderos dispositivos móviles. Estas cuatro características son:

- 1) Movilidad
- 2) Tamaño reducido
- 3) Comunicación inalámbrica
- 4) Interacción con las personas

## **2.7 Contribución Antecedentes**

En la sección 1.2 se describen los antecedentes que sirven como soporte y aportes para el desarrollo de la arquitectura propuesta en el presente trabajo de investigación. Cada uno de ellos allí descritos contribuye en algún elemento que se describe en la siguiente tabla:

Fermi y Otros (2012)	El flujo de trabajo que ellos proponen, comenzando con imágenes médicas obtenidas de diferentes fuentes (tomografía computarizada, ultrasonido, resonancia magnética, entre otros) que se encuentran en formato DICOM y se envían a los PACS. Los usuarios utilizan estaciones de trabajo DICOM con capacidad de obtener las imágenes de los PACS para su posterior visualización o diagnóstico en el sistema móvil.
Ojog y Arias (2012)	Ellos alojan el archivo DICOM en el lado del servidor y entrega un modelo dinámico X3D al dispositivo móvil para la visualización e interacción con el usuario. El procesamiento se lleva a cabo en la nube y la representación 3D se realiza en el dispositivo móvil. Su plataforma se basa en una arquitectura cliente-servidor que permite a los médicos o radiólogos visualizar, analizar e interactuar con la información del paciente almacenada en un repositorio DICOM web. A pesar que no se utiliza modelos X3D ni HTML5 en la arquitectura presentada en este proyecto de tesis, permitió tomar una decisión de realizar el procesamiento en el dispositivo móvil, por lo complejo del procesamiento en la nube e integrar otros elementos del sistema radiológico y del sistema de información del centro hospitalario. Además la plataforma cliente-servidor es la más adecuada para este tipo de arquitectura, por motivos de seguridad, fiabilidad y desempeño.
Goretzki (2014)	En su investigación se señalan varias técnicas tanto para la superficie y renderización volumétrica basado en la API de JAVA 3D lo cual permite tomar la decisión de usar modelos STL para la visualización en el dispositivo móvil para que el usuario sea capaz de interactuar con él, pudiendo girar, mover o escalar en el interior del espacio 3D.
Ramírez y Coto (2012)	Ellos resaltan la importancia de poseer un repositorio de modelos 3D para almacenar de forma segura cada estudio efectuado al paciente, de igual forma, señalan la importancia que usuario pueda interactuar con dicho estudio, con cada acción de rotación, movimiento, zoom, entre otros.

**Tabla 2.** *Contribución de los antecedentes*

## **CAPÍTULO III**

### **METODOLOGÍA**

La investigación, de acuerdo con Sabino (1992), se define como “un esfuerzo que se emprende para resolver un problema, claro está, un problema de conocimiento”. Para emprender un trabajo investigación es necesario tomar en cuenta los elementos básicos en la formulación y desarrollo del proyecto, para ello, la metodología hace referencia a un conjunto de pasos racionales para alcanzar uno o varios objetivos dentro del proceso de investigación ajustándose a las necesidades y objetivos que se cuentan a lo largo del ciclo de vida.

La investigación se puede clasificar de diferentes maneras según su nivel, diseño y propósito, lo que conlleva al tipo de investigación y dependiendo de ésta, determinará los pasos a seguir en el estudio y la forma de recoger los datos o información. Además, se van a utilizar elementos de metodologías investigativas y elementos para el desarrollo de la plataforma tecnológica o arquitectura mediante el método espiral de Bohem que se explica más adelante.

#### **3.1 Tipo de investigación**

De acuerdo con Padrón (1998) las Investigaciones Aplicativas

Parten del hecho de que, dentro de la secuencia de trabajo de la Línea, existen teorías cuya verosimilitud ha aumentado gracias a un cierto número de contrastaciones y, además, del hecho de que en el mundo de las necesidades de desarrollo existen requerimientos que pueden ser satisfechos aprovechando esas teorías. Su objetivo central está en proveer tecnologías o esquemas de acción derivados de los conocimientos teóricos construidos dentro de la secuencia de la Línea. Estas investigaciones carecen, propiamente hablando, de preguntas. Más bien tienden a establecer una relación productiva, ingeniosa y creativa, entre las



posibilidades de un modelo teórico, por un lado, y las dificultades o necesidades que se confrontan en el terreno de la práctica, por otro lado.

Según lo antes expuesto, el tipo de investigación del presente trabajo de grado es aplicativa, ya que implica proveer tecnologías derivadas de los conocimientos teóricos; es decir, una implementación tecnológica para el desarrollo de una arquitectura para el despliegue 3D en dispositivos móviles de imágenes tomadas de un tomógrafo.

### **3.2 Diseño de la investigación**

El diseño de la investigación permite determinar la estrategia para resolver el problema. Sabino (1992) señala que si los datos que se van a emplear han sido recolectados en otras investigaciones y son conocidos mediante informes correspondientes pertenece a un diseño bibliográfico, ya que han sido obtenidos y procesados por otros autores para con otros fines determinados. Dichas informaciones proceden siempre de documentos escritos, pues esa es la forma uniforme en que se emiten los informes científicos.

Con base en la situación descrita, la presente investigación presenta un diseño bibliográfico, ya que los datos que se usarán serán obtenidos de fuentes secundarias, es decir, de fuentes impresas y/o electrónicas.

### **3.3 Técnicas e instrumentos**

Una entrevista no estructurada es aquella donde existe un alto margen de libertad para formular las preguntas y respuestas. Por lo tanto no se guía por un cuestionario o modelo rígido, sino por un cierto grado de espontaneidad, mayor o menor según el tipo de entrevista (Ander 1972, Citado En sabino 1992).

Asimismo Sabino (1992) explica: “Los datos secundarios, por otra parte, son registros escritos que proceden también de un contacto con la práctica, pero que ya han sido recogidos y muchas veces procesados por otros investigadores.”

En éste proyecto de investigación se contará con una técnica de recolección de tipo revisión documenta y entrevista no estructurada con los expertos, ya que los datos e información requeridos para la consecución de los objetivos se obtendrán de planteamientos y estudios plasmados en documentos impresos y vía web. La técnica a utilizar no será estrictamente constituida, se tendrá un cuadro registro para organizar las consultas y los datos de interés para la el desarrollo de los objetivos del proyecto.

Este modelo se desarrollará con el fin de que cada uno de los objetivos para la implementación del desarrollo de la arquitectura para el despliegue 3D en dispositivos móviles de imágenes tomadas de un tomógrafo, se cumpla satisfactoriamente logrando un producto de aplicación real y ajustado a los lineamientos de los expertos y apegados a los estudios realizados en el área.

### **3.4 Metodología de desarrollo**

El método empleado en el presente trabajo de grado se encuentra estructurado en dos grandes partes, una referida a la investigación documental de trabajos y arquitecturas similares a la propuesta, el uso de herramientas y tecnología para el desarrollo de aplicaciones y/o prototipos equivalentes y una segunda fase dedicada al desarrollo programado de la arquitectura del sistema propuesto.

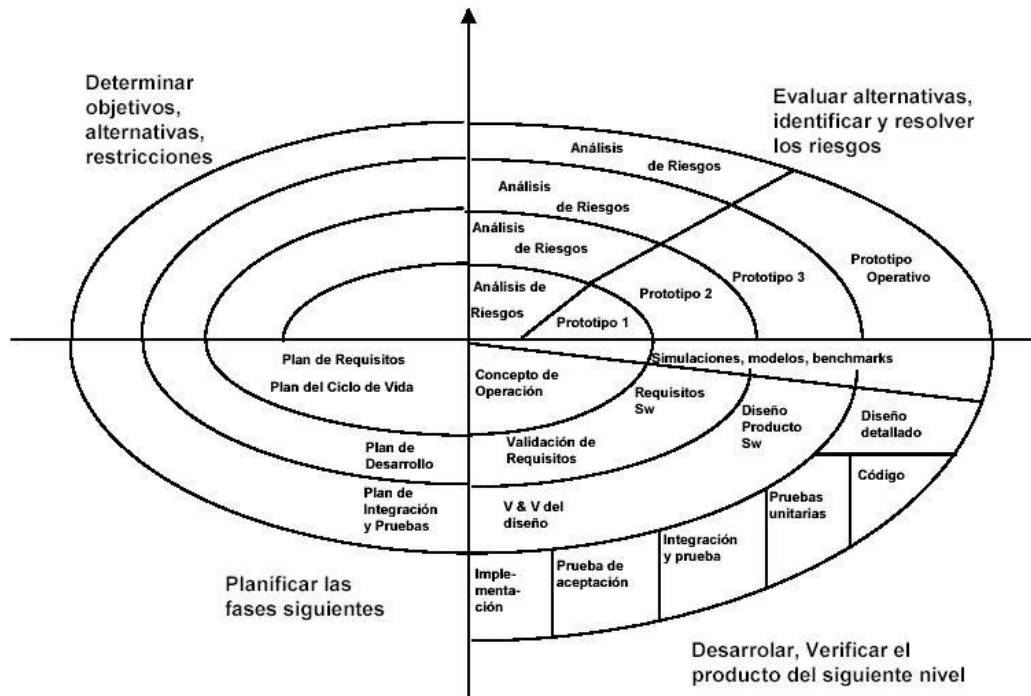
Durante el desarrollo de cada elemento compuesto dentro de la arquitectura se utiliza la metodología de Boehm (1998) (Figura 9) que es un modelo en espiral basado en el enfoque evolutivo, creando prototipos funcionales a los cuales se les añaden características en cada ciclo evolutivo dado a que hay que efectuar un análisis de riesgos según las características del dispositivo móvil, el tamaño del modelo 3D, los requerimientos de red, entre otros riesgos técnicos en todas las etapas, permitiendo así generar el prototipo básico para cumplir con el

objetivo general, además que puede adaptarse y aplicarse a lo largo de la vida del software de computadora.

Boehm (1998) señala que el método se encuentra dividido en cuatro cuadrantes o fases, entre ellas la Determinación de objetivos, alternativas y restricciones, Evaluación de alternativas e identificación y resolución de riesgos, Desarrollo, verificación del producto y Planificación en una estructura de iteraciones que permite integrar cada una de ellas. Para cumplir cada una de ellas es requerido los siguientes pasos:

- **Determinación de objetivos, alternativas y restricciones:** en esta fase se identifican los objetivos, alternativas y restricciones basándose en los requerimientos, especificaciones y restricciones.
- **Evaluación de alternativas e identificación y resolución de riesgos:** en este momento se realiza el análisis de riesgo y se toman las decisiones sobre las alternativas para el desarrollo del prototipo.
- **Desarrollo, verificación del producto:** se lleva a cabo el desarrollo del producto y las pruebas consiguientes, también se puede realizar simulaciones, modelos entre otras pruebas con la finalidad de verificar que el producto cumpla con los objetivos planteados.
- **Planificación:** se hace una revisión de los objetivos alcanzados, se realiza el plan de implementación y el desarrollo para determinar si se ha de continuar con el producto.

La Figura 12 ilustra las fases que comprenden la metodología Boehm expuesta:



**Figura 12:** Modelo en espiral de Boehm (Tomado de Santos, 2014)

## CAPÍTULO IV

### DISEÑO DE LA ARQUITECTURA

El diseño de la arquitectura propuesta se desarrolla siguiendo las fases del método de Boehm descrito en el capítulo anterior. Durante cada iteración se establecen los requisitos para cumplir los objetivos de dicha arquitectura.

#### 4.1 Determinación de los objetivos

La determinación de los objetivos conlleva a la necesidad que debe cubrir la plataforma de desarrollo, tomando en cuenta las alternativas o las diferentes formas de conseguir el objetivo general de forma exitosa, lo cual lleva al establecimiento de los objetivos específicos.

- Analizar la captura de datos generados por los tomógrafos.
- Analizar el proceso de despliegue tridimensional en dispositivos móviles.
- Diseñar la arquitectura básica para cada uno de los componentes requeridos para el funcionamiento del prototipo.
- Desarrollar un prototipo básico para el despliegue tridimensional de archivos *Stereolithography* (STL) sobre la arquitectura propuesta.
- Realizar las pruebas de rendimiento sobre el prototipo desarrollado.
- Establecer las características básicas mínimas de red y del dispositivo móvil para el despliegue del prototipo.

De acuerdo con el *Project Management Institute* (2004) durante el desarrollo del proyecto puede surgir un cierto evento o condición incierta pudiera llevar a cabo un efecto positivo o negativo en al menos uno de los objetivos planteados, por lo tanto es importante determinar los riesgos que pueden ocurrir durante el desarrollo de la arquitectura para actuar de forma

proactiva o mitigar sus posibles consecuencias. Entre los posibles riesgos se pueden mencionar:

- Aumento en el tiempo de desarrollo del prototipo.
- Incompatibilidad en la interconexión entre los sistemas, versiones y tecnologías que forman parte de la arquitectura.
- Tiempo de aprendizaje sobre el manejo de la tecnología para crear el prototipo y los formatos de las imágenes DICOM
- Bajo rendimiento o largos tiempos de respuesta en el despliegue tridimensional.
- Alto costo de los dispositivos móviles de alta gama.
- Diseño inadecuado de la arquitectura

## **4.2 Fase de diseño**

Esta fase toma en cuenta los requerimientos del sistema en donde se detalla la arquitectura funcional, los procesos y estructuras que lo componen.

### **4.2.1 Conceptualización.**

Se definen los requisitos funcionales y no funcionales de la arquitectura propuesta.

#### **4.2.1.1 Requisitos Funcionales:**

Los requisitos funcionales son aquellos que definen una función del sistema de software, sus entradas, comportamiento y salida. La arquitectura para el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos necesita de los requisitos que se determinan en tres iteraciones, una para la unidad de radiología donde el personal especializado debe poseer la infraestructura completamente funcional para poder generar el estudio tridimensional del área de interés del paciente a través de las imágenes DICOM dadas por el tomógrafo; otra mediante un sistema web que permita cargar el resultado del estudio para su

posterior consulta y por último un prototipo donde el mismo especialista consulta dichos resultados y modelo 3D. A continuación se describe cada uno de ellos:

<b>Requisito Funcional</b>	<b>Descripción</b>
<b>Generar Modelo 3D del Estudio</b>	Permite generar el modelo tridimensional del área de interés según la cantidad de cortes tomadas por el tomógrafo al paciente
<b>Cargar Modelo 3D Repositorio</b>	El Radiólogo debe colocar de forma digital el modelo tridimensional en un repositorio donde se encuentren organizados cada uno de ellos
<b>Cargar Examen Paciente</b>	Permite cargar la información básica del examen médico para su posterior consulta.
<b>Ingresar Sistema</b>	Permite ingresar el sistema web con las credenciales dadas para proceder a efectuar la carga, edición, modificación o eliminación de los exámenes y sus respectivos datos asociados
<b>Gestionar Examen Médico</b>	Permite gestionar los estudios médicos efectuados al paciente, permitiendo agregar, editar, consultar o eliminar los mismos.
<b>Gestionar Paciente</b>	Gestiona los pacientes a quienes se les efectúa los estudios médicos, permitiendo agregar, editar, consultar o eliminar los mismos.
<b>Gestionar Médico</b>	Gestiona los médicos especialistas quienes tendrán acceso al estudio 3D, permitiendo agregar, editar, consultar o eliminar los mismos.
<b>Gestionar Especialidad</b>	Permite agregar, editar, consultar o eliminar las especialidades de los médicos.
<b>Gestionar Tipo Examen</b>	Permite agregar, editar, consultar o eliminar los tipos de estudio que se efectúan sobre la región de interés.
<b>Gestionar Persona</b>	Permite agregar, editar, consultar o eliminar las personas del sistema, entre ellos se encuentran los médicos, los pacientes y los usuarios y el cual incluye el género, país y estado civil.
<b>Gestionar Usuarios</b>	Permite agregar, editar, consultar o eliminar los usuarios del sistema, ya sean los que tienen acceso al prototipo móvil o quienes hacen la carga de los estudios al sistema web.
<b>Ingresar Credenciales</b>	Permite acceder al prototipo del sistema móvil
<b>Seleccionar Estudio Paciente</b>	Permite seleccionar el estudio del paciente entre el listado de cada uno de ellos
<b>Visualizar Modelo 3D Estudio</b>	Permite desplegar el modelo 3D para mover, rotar, acercar o alejar la región de interés.
<b>Cerrar Sesión</b>	Es necesario que al finalizar de consultar los estudios se cierre la sesión y borren los archivos temporales del sistema.
<b>Cambiar Contraseña</b>	Por medidas de seguridad se requiere que el usuario del

	prototipo móvil pueda modificar sus credenciales de acceso.
--	---

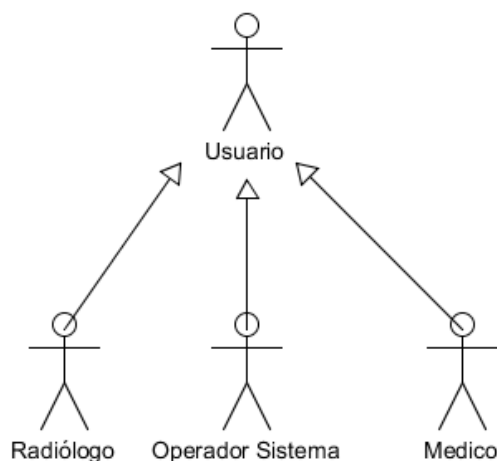
**Tabla 3.** *Requisitos Funcionales*

#### 4.2.1.2 Casos de uso

Los casos de uso describen los pasos a seguir para llevar a cabo algún proceso y se encuentran relacionados con los requisitos funcionales mencionados anteriormente, el cual poseen roles que son efectuados por los actores que interactúan en la plataforma propuesta. A continuación se muestran los casos de uso siguiendo el esquema de las tres iteraciones para la metodología dada.

##### *a. Diagrama de Actores:*

Antes de hacer la representación de los caso de uso, se describen los actores que se encuentran en la plataforma y que corresponden a los usuarios que interactúan con los elementos que la integran, entre ellos está el Radiólogo, el Operador del Sistema y el Médico, cada uno de ellos tienen ciertas responsabilidades que se explican más adelante en cada caso de uso. La siguiente Figura muestra el modelado de los actores:

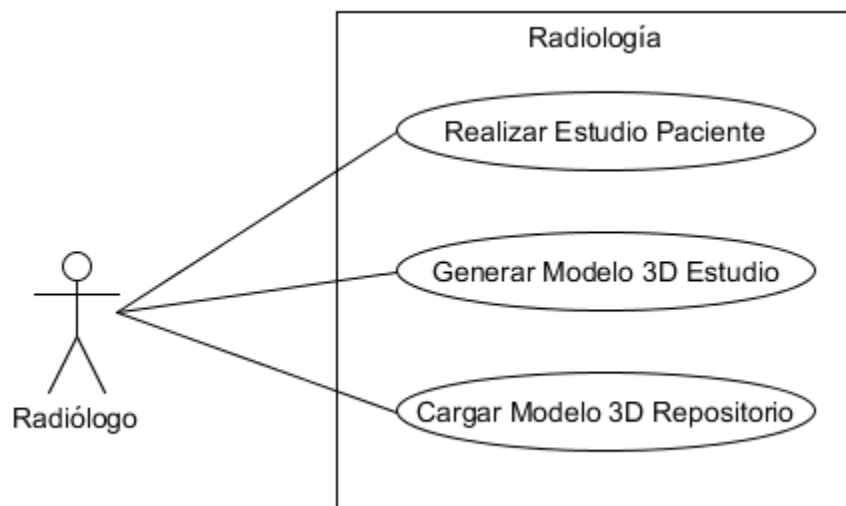


**Figura 13:** *Diagrama de Actores*



**b. Caso de uso unidad de radiología:**

En la Figura 14 se muestra el caso de uso en la unidad de radiología, su actor y cada uno de sus procesos:



**Figura 14:** Caso de uso radiólogo

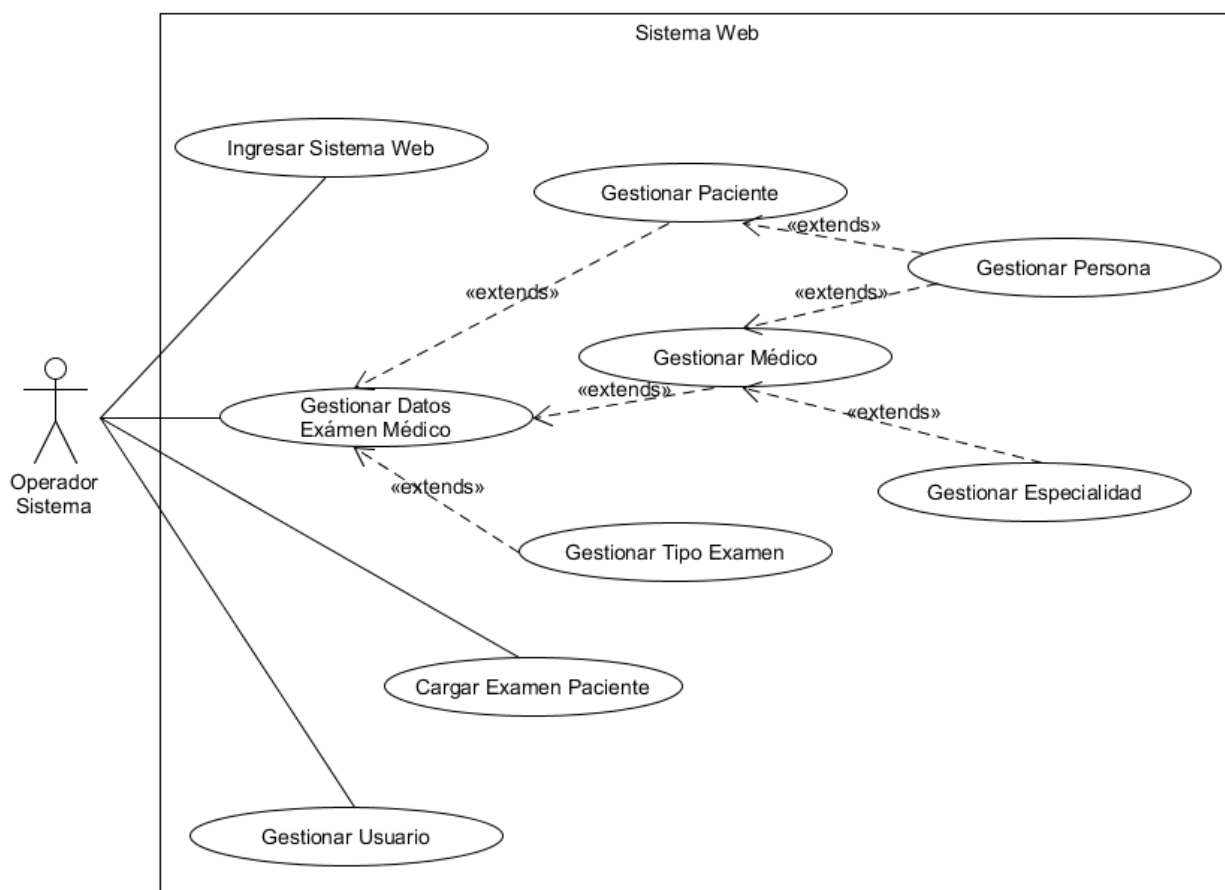
Actor	Rol	Descripción
Radiólogo	<b>Realizar Estudio Paciente</b>	El Radiólogo comienza por efectuar el estudio en la región del interés para el paciente en el tomógrafo
	<b>Generar Modelo 3D Estudio</b>	Este proceso permite generar el modelo 3D a través del software del visualizador DICOM dentro del sistema de archivado y transmisión de imágenes (PACS)
	<b>Cargar Modelo 3D Repositorio</b>	Permite cargar el estudio a un repositorio de exámenes, el Radiólogo debe almacenar el archivo STL a una carpeta

		compartida en el sistema operativo.
--	--	-------------------------------------

**Tabla 4.** Descripción Caso de Uso Unidad de Radiología

**c. Caso de uso sistema web:**

En la Figura 15 se muestra el caso de uso del sistema web, sus actores y cada uno de sus procesos.



**Figura 15:** Caso de uso sistema web

Actor	Rol	Descripción
<b>Operador del Sistema</b>	<b>Ingresar Sistema</b>	Se ingresan las credenciales para poder acceder al sistema web, en caso de no poseerlas debe hacer el registro del usuario y calve por la base de

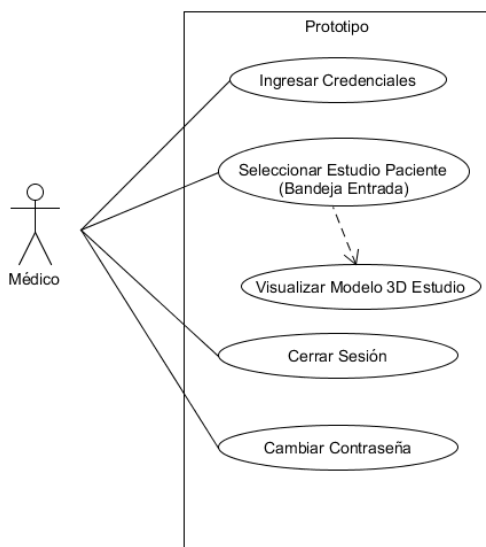
		datos
	<b>Gestionar Examen Médico</b>	Permite insertar, modificar, eliminar o consultar los exámenes médicos generados por el radiólogo a través de los PACS y el visualizador DICOM
	<b>Gestionar Paciente</b>	Permite insertar, modificar, eliminar o consultar los pacientes de los estudios
	<b>Gestionar Médico</b>	Permite insertar, modificar, eliminar o consultar los médicos encargados de revisar el estudio.
	<b>Gestionar Especialidad</b>	Permite insertar, modificar, eliminar o consultar las especialidades de los médicos.
	<b>Gestionar Tipo Examen</b>	Permite insertar, modificar, eliminar o consultar los tipos de exámenes realizados
	<b>Gestionar Persona</b>	Permite insertar, modificar, eliminar o consultar las personas del sistema, ya sean médicos, pacientes o usuarios
	<b>Gestionar Usuario</b>	Permite insertar, modificar, eliminar o consultar los usuarios del sistema
	<b>Cargar Examen Paciente</b>	El operador ingresa al sistema web para gestionar el examen del paciente dado, mediante las credenciales entregadas por el administrador del

		sistema, en caso de no poseerlas, debe comunicarse con el responsable del sistema web para hacer la solicitud
--	--	---

**Tabla 5.** Descripción Caso de Uso Sistema Web.

**d. Caso de uso Prototipo móvil:**

En la Figura 16 se muestra el caso de uso del prototipo móvil, sus actores y cada uno de sus procesos, de las cuales el usuario (en este caso el mismo médico) debe ingresar las credenciales otorgadas por el administrador del sistema, es decir, el nombre de usuario y clave, que en cualquier momento puede modificar por su propia cuenta accediendo al menú de cambiar contraseña. Una vez que el prototipo valida las credenciales almacenadas en la base de datos centralizada, el médico puede seleccionar el estudio en la llamada bandeja de exámenes, donde se descargan al dispositivo todos los exámenes activos de sus pacientes, pudiendo seleccionar cualquiera de ellos, para ser descargados a un archivo temporal del dispositivo y hacer el despliegue 3D para hacer el análisis tomar las decisiones requeridas. Al finalizar el uso del prototipo, puede el usuario cerrar la sesión para limpiar la caché del dispositivo y cerrar la conexión al sistema.



**Figura 16:** Caso de uso prototipo móvil

Actor	Rol	Descripción
<b>Medico</b>	<b>Ingresar Credenciales</b>	Este proceso permite ingresar las credenciales del médico especialista para acceder al prototipo del sistema móvil, en caso de no poseerlas debe hacer la solicitud al administrador del sistema
	<b>Seleccionar Estudio Paciente (Bandeja Entrada)</b>	El prototipo despliega un listado con todos los estudios dados al médico, donde mismo debe seleccionar cual desea desplegar en la pantalla. En caso de no encontrarse el estudio, debe comunicarse con el administrador del sistema
	<b>Visualizar Modelo 3D Estudio</b>	El medico puede visualizar o desplegar en la pantalla del dispositivo el estudio, permitiendo rotar, mover, acercar o alejar el área interesada.
	<b>Cerrar Sesión</b>	Permite cerrar la sesión en el dispositivo y limpiar la memoria temporal para que de esa forma no se acumulen archivos.
	<b>Cambiar Contraseña</b>	Permite modificar la clave de acceso para tener mayor seguridad ya que los estudios son información confidencial que no todo el personal

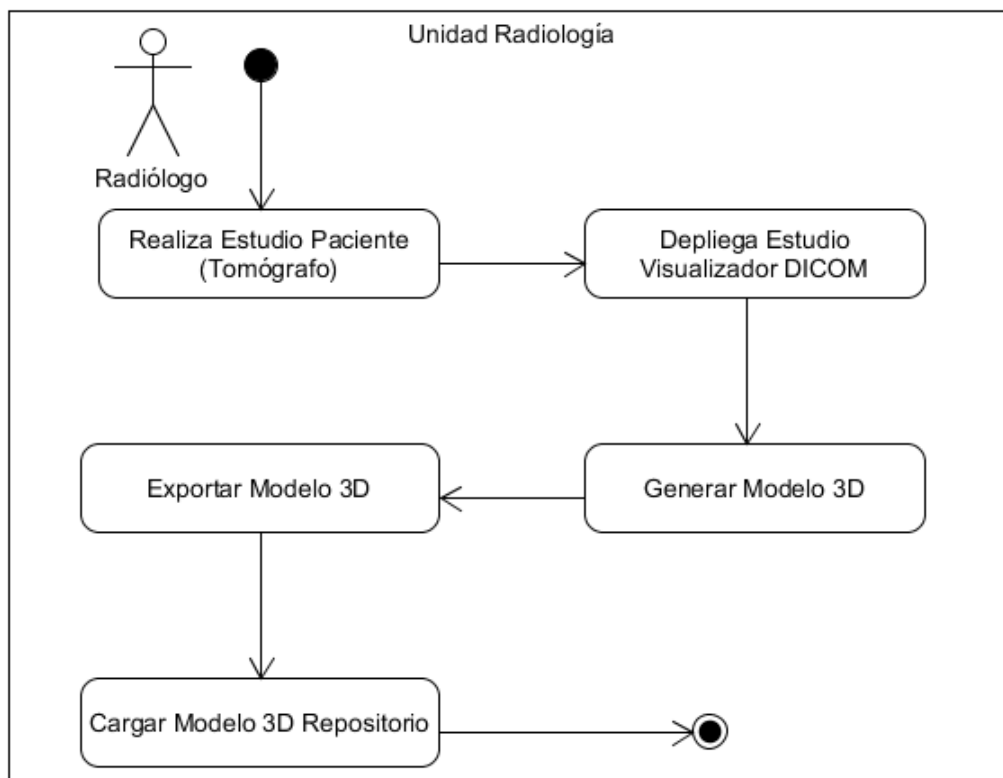
		debería tener acceso.
--	--	-----------------------

**Tabla 6:** *Descripción Caso de Uso Prototipo Móvil.*

#### **4.2.1.3 Diagrama de actividades:**

##### ***a. Unidad de Radiología:***

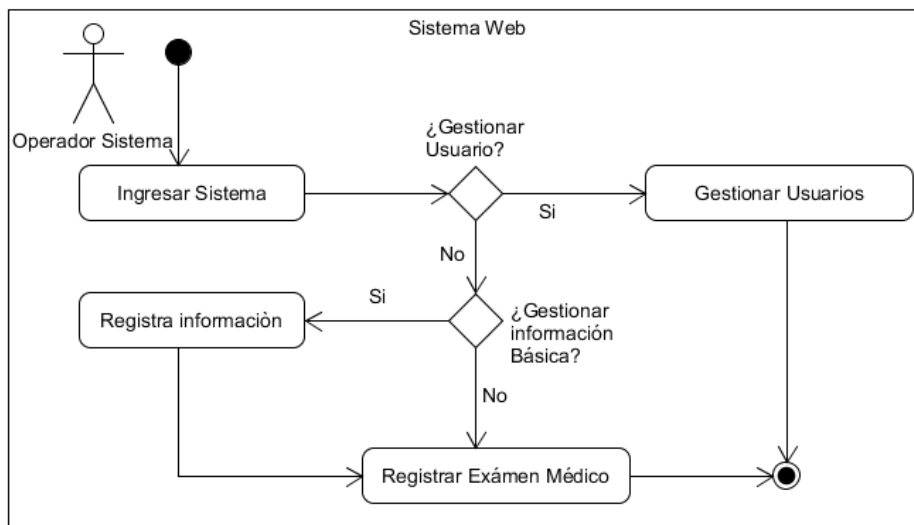
En la Figura 17 se muestra el diagrama de actividades para la unidad de radiología. Es importante mencionar que el radiólogo realiza el estudio al paciente a través del tomógrafo y utilizando el software visualizador DICOM para generar el modelo de la región de interés y exportarlo para su posterior carga a un repositorio centralizado que tiene por objetivo organizar, archivar, preservar y difundir los modelos 3D en formato .STL (Stereolithography) donde se almacenan en un directorio cada uno de los resultados de los estudios de los pacientes para su posterior despliegue en los dispositivos móviles. El repositorio permite a los usuarios del sistema web tener y del prototipo tener acceso a los estudios que han sido subidos (por medio de una carpeta compartida) a carpetas designadas en el servidor. Un repositorio puede estar organizado para todo el personal encargado de generar el estudio; puede ser dividido en carpetas para diferentes departamentos o áreas de trabajo o para un usuario individual, en otras palabras, la organización interna del repositorio de archivos es dada según las necesidades de la organización donde se encuentre la arquitectura propuesta. Posteriormente se debe ingresar al sistema web donde se carga (en el caso de no existir) el estado civil de las personas registradas, el género de las personas, los médicos, pacientes, países para identificar las nacionalidades, las personas, el tipo de especialidad para los médicos, el tipo de examen o estudio y los usuarios del sistema para por último registrar el examen del paciente.



**Figura 17:** Diagrama de Actividades Unidad Radiología

**b. Sistema Web:**

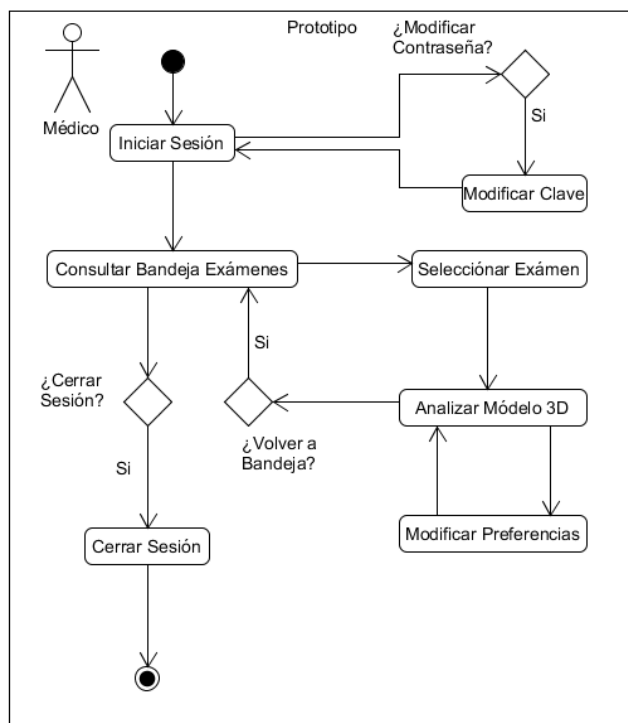
En la Figura 18 se muestra el diagrama de actividades del sistema web donde el radiólogo tiene la posibilidad de agregar, eliminar, modificar o consultar los usuarios del sistema y de gestionar la información básica para los resultados (estado civil, paciente, medico, especialidad, tipo de estudio, entre otros) que son requeridas para el examen médico.



**Figura 18:** Diagrama de Actividades Sistema Web

**c. Prototipo Móvil:**

La Figura 19 se muestra el diagrama de actividades para el prototipo móvil.



**Figura 19:** Diagrama de Actividades prototipo móvil

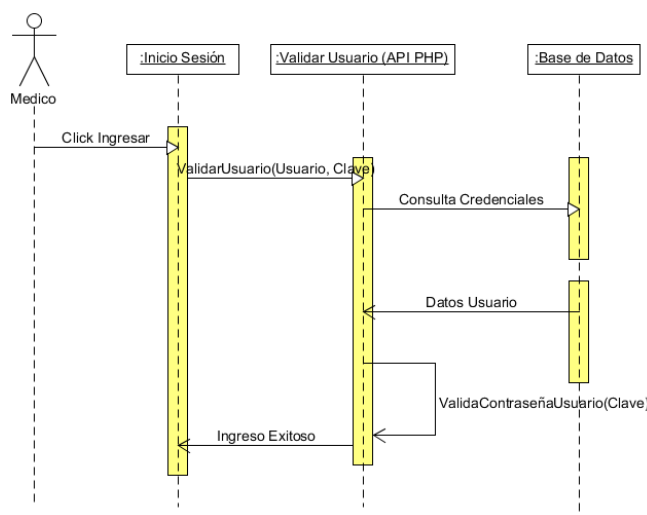


#### 4.2.1.4 Diagrama de Secuencia para el prototipo móvil

De manera más específica se presenta la descripción de los procesos básicos del prototipo, comenzando por el inicio de sesión:

##### a. Proceso de inicio de sesión

El médico (usuario) debe ingresar las credenciales en a interfaz de inicio de sesión y presionar el botón de ingresar, el prototipo mediante una lógica de programación se encarga de validar el usuario y clave conectándose a la API PHP que se encuentra en la red, esta a su vez hace una consulta a la Base de Datos, retornando la contraseña para dicho usuario que es comparada y validada por la misma API para determinar si se debe otorgar o no acceso a los estudios de los pacientes. En el siguiente diagrama (Figura 20) de secuencia se muestra el procedimiento.

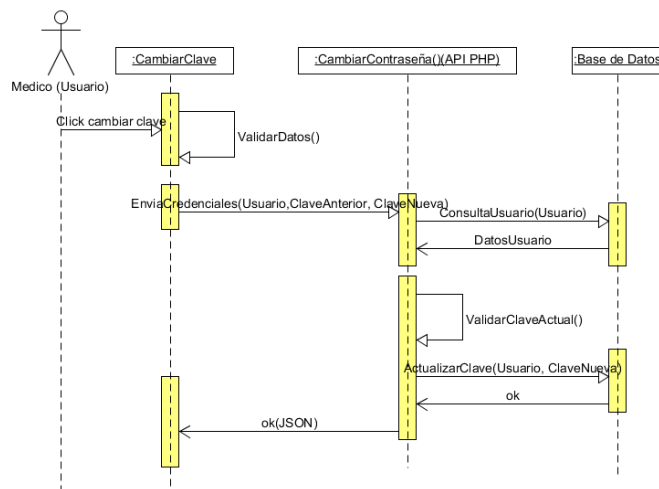


**Figura 20:** Diagrama de secuencia inició de sesión

##### b. Proceso cambiar contraseña

En la siguiente Figura 21 se encuentra el diagrama de secuencia que se debe llevar a cabo cuando un usuario desea modificar su contraseña, para ello debe llenar el formulario que contiene el nombre de usuario, la contraseña actual y la nueva contraseña, luego presionar el botón de cambiar clave, la interfaz de cambiar clave valida si la nueva contraseña es correcta, en

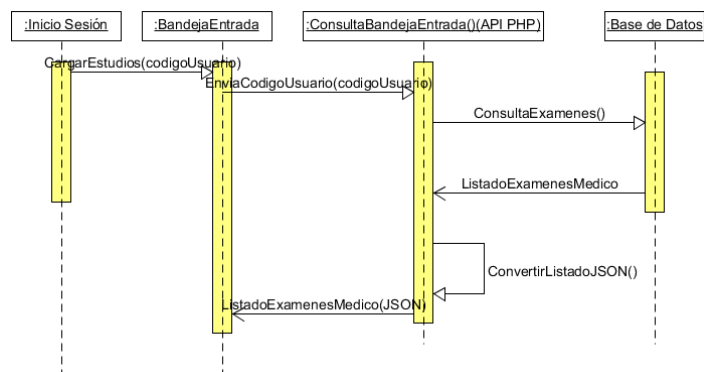
caso afirmativo, envía las credenciales a la API PHP, donde el método correspondiente hace una consulta a la Base de Datos para determinar que el usuario es quien dice ser, en caso afirmativo, se hace una actualización sobre el registro de la base de datos y se retorna en formato JSON la respuesta a la interfaz si fue o no modificada la contraseña.



**Figura 21:** Diagrama de secuencia cambiar contraseña

### c. Proceso para cargar la bandeja de exámenes

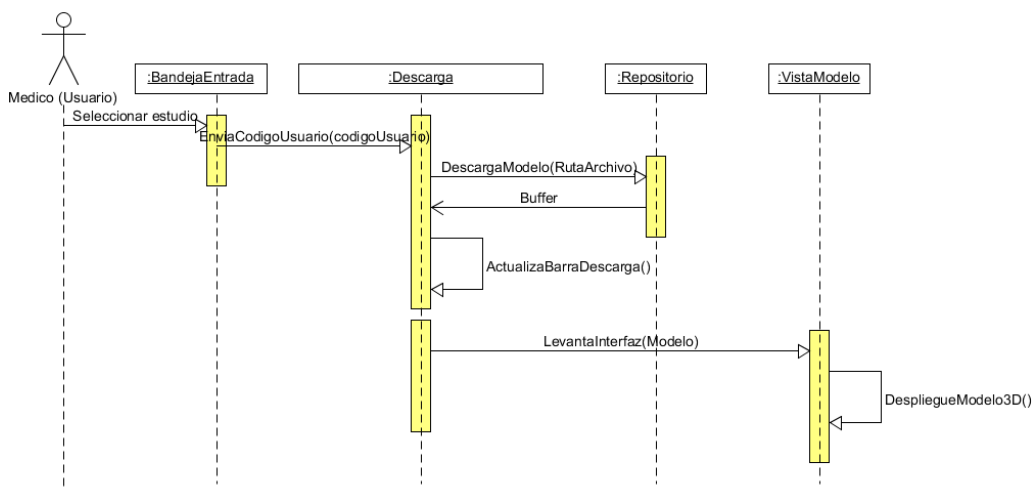
Luego que el usuario inicia la sesión se despliega una interfaz llamada Bandeja de Exámenes en donde se descargan todos los estudios activos de los pacientes por parte del médico. Al crear la interfaz, automáticamente se hace una consulta a la API PHP con el código del usuario para poder consultar en la Base de Datos todos sus exámenes. La consulta genera un listado de todos los exámenes activos de los pacientes, para luego ser transformados a un Array en formato JSON y ser desplegados en la Bandeja de Exámenes. En caso de no existir ningún examen activo, la Bandeja de Exámenes estaría vacía. La siguiente Figura 22 muestra el proceso descrito.



**Figura 22:** Diagrama de secuencia para cargar la Bandeja de Exámenes

#### d. Proceso para desplegar modelo 3D

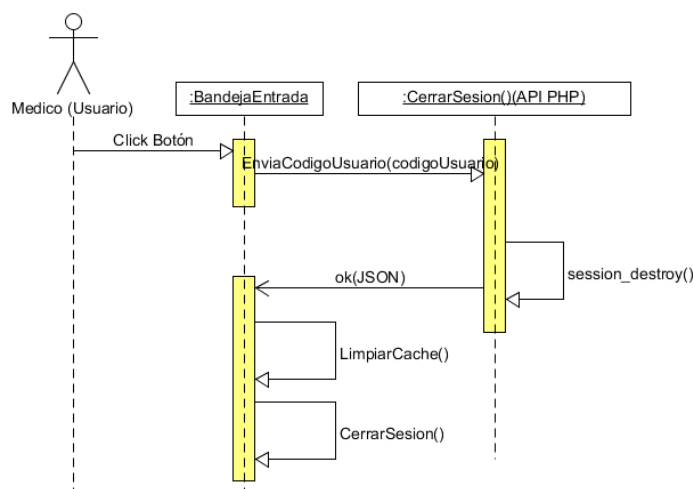
En la siguiente Figura 23 se encuentra el diagrama de secuencia que se debe llevar a cabo cuando el médico o usuario desea cargar el modelo tridimensional en su dispositivo móvil. Una vez iniciada la sesión y cargada la Bandeja de Exámenes, el médico selecciona el estudio que desea hacer el diagnóstico en la propia interfaz, esta a su vez se conecta al repositorio mediante otra interfaz de descarga con una barra de progreso que indica el porcentaje de bytes descargados según el tamaño del archivo, allí se ejecuta un ciclo mediante un hilo hasta que se culmine de transferir el archivo completo. Al finalizar, se levanta la interfaz de Vista del Modelo donde el usuario puede mover y enfocar el mismo para su estudio.



**Figura 23:** Diagrama de secuencia para desplegar modelo 3D

### e. Proceso cerrar sesión

En la siguiente Figura 24 se encuentra el diagrama de secuencia que se debe llevar a cabo cuando el médico cierra la sesión en el prototipo. El botón se encuentra en la interfaz de Bandeja de Exámenes, al presionar el botón, se envía mediante el método de cerrar sesión el código del usuario a la API PHP, allí se llama a la función *sesión\_destroy()* que permite destruir toda la información registrada de la sesión asociada a la conexión establecida por el mismo usuario. Posteriormente la API retorna en formato JSON la respuesta de sesión destruida exitosamente para que en la misma interfaz se limpie la cache ocupada por la descarga de los archivos temporales correspondientes a los modelos es estudio entre otros y se culmine la sesión cerrando el prototipo del dispositivo móvil.



**Figura 24:** Diagrama de secuencia cerrar sesión

#### 4.2.1.5 Repositorio de modelos

La arquitectura propuesta contiene un repositorio centralizado que tiene por objetivo organizar, archivar, preservar y difundir los modelos 3D en formato .STL (Stereolithography) donde se almacenan en un directorio cada uno de los resultados de los estudios de los pacientes para su posterior despliegue en los dispositivos móviles. El repositorio permite a los usuarios del sistema web tener y del prototipo tener acceso a los estudios que han sido en el servidor.

#### 4.2.1.6 Diagrama de paquetes del prototipo propuesto

Para describir de manera general como se encuentra estructurado el código fuente del prototipo se presenta el diagrama de paquetes (Figura 25) que muestra la forma en que se encuentra dividido el código en agrupaciones lógicas y como entre ellas se relacionan. Entre los paquetes que allí se describen, se encuentra el paquete principal *ula*, luego *com* y *adtviewer* (*ula.com.adtviewer*) de allí se divide en los paquetes con las clases necesarias para las utilidades, los objetos, el renderizado, la vista, las actividades y una librería con clases requeridas para la configuración y funcionamiento del mismo.

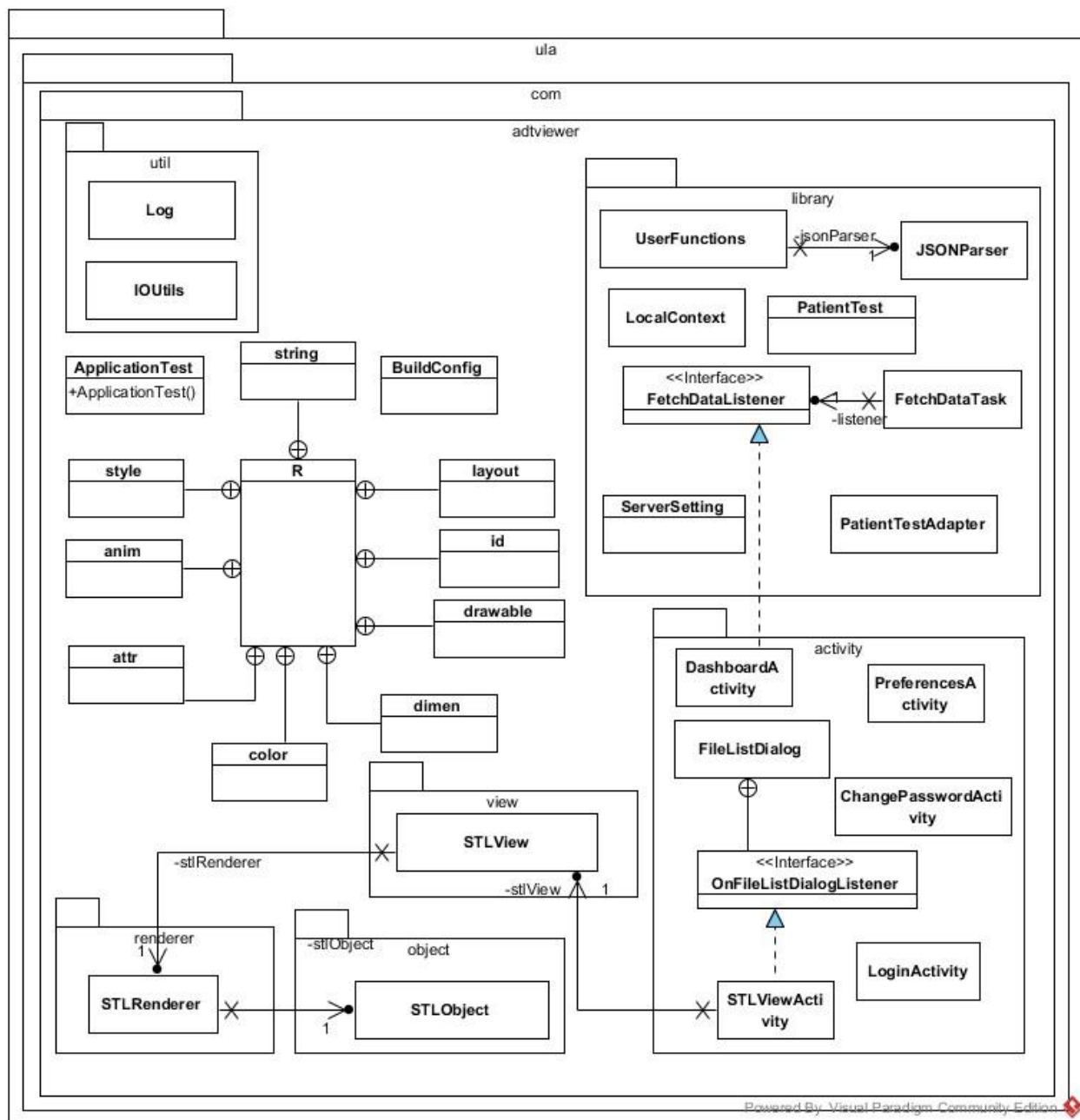


Figura 25: Diagrama de paquetes prototipo propuesto

#### 4.2.1.7 Diagrama de clases del prototipo propuesto

Cada uno de los paquetes descritos en el diagrama de paquetes contiene una serie de clases orientadas a objetos requeridas para el funcionamiento del prototipo, su estructura se

define en el siguiente diagrama de clases (Figura 26) donde se muestran sus relaciones, atributos y métodos en cada una de ellas.

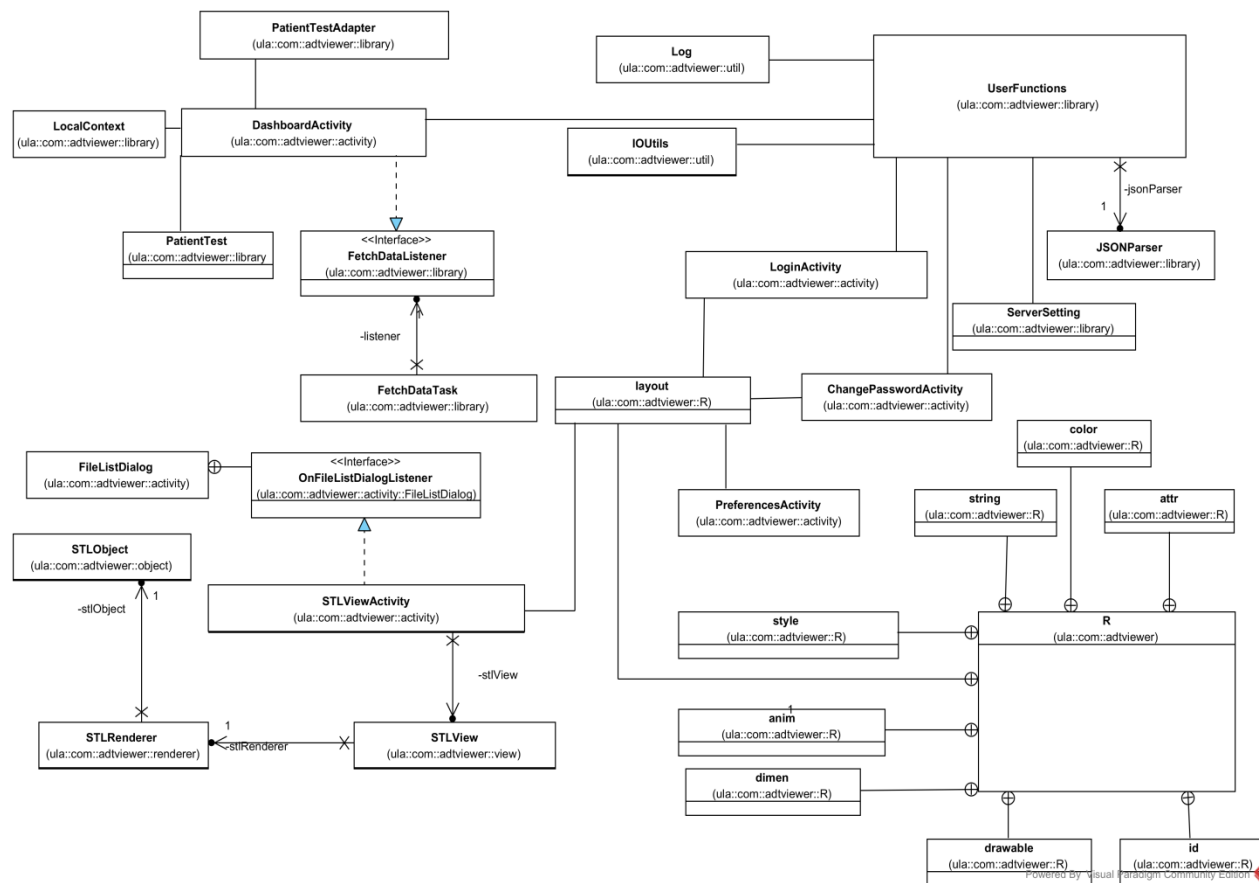
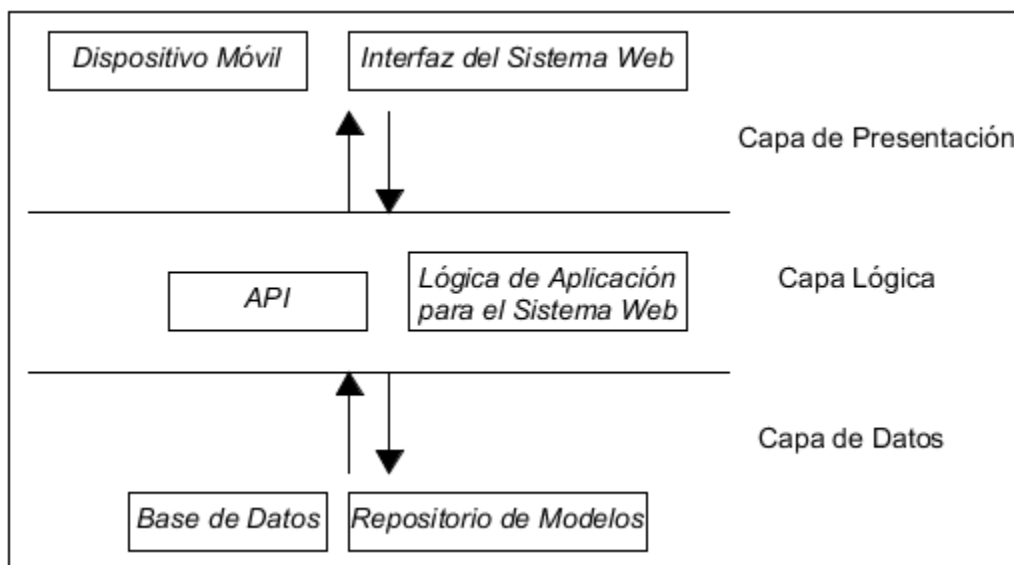


Figura 26: Diagrama de clases del prototipo

#### 4.2.1.8 Visión Arquitectónica del Sistema

El tipo de arquitectura presente en la plataforma propuesta corresponde a un modelo multicapa de tres capas: la **capa de presentación** que es el nivel más alto donde se encuentra la interfaz del usuario correspondiente al prototipo del dispositivo móvil y a las pantallas del sistema web; la **capa lógica** que coordina y procesa los comandos dados por el prototipo y el sistema web, además intercambia información entre las dos capas adyacentes; y la **capa de datos** donde almacena y recupera la base de datos y el repositorio de modelos donde se encuentra la

información que es entregada a la capa lógica y eventualmente al usuario final. La Figura 27 se explica en forma de diagrama el modelo de tres capas explicado.

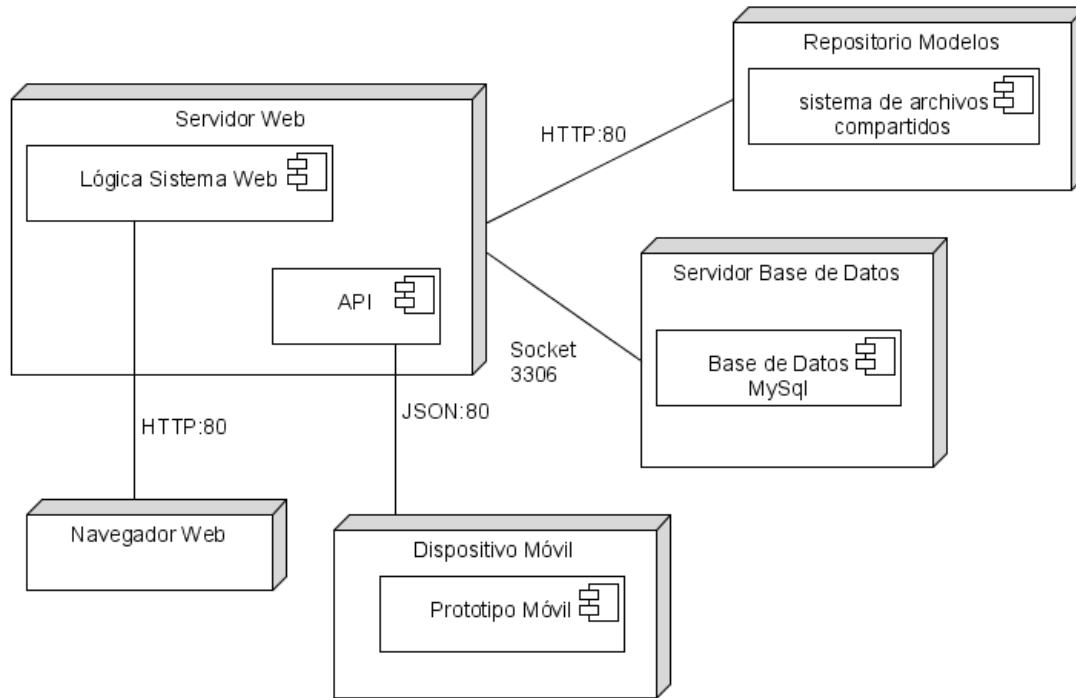


**Figura 27:** *Visión arquitectónica del sistema*

#### 4.2.1.9 Diagrama de Despliegue

El diagrama de despliegue modela las relaciones físicas de la arquitectura física en tiempo de ejecución, dado que es un sistema de tres capas, por una parte se tiene el dispositivo móvil y el navegador web correspondiente a la capa de presentación, por otra parte está el servidor de repositorios dado por el sistema de archivos compartidos y el servidor de base de datos que utiliza el motor Mysql. Por último se encuentra el servidor web donde se encuentra el código fuente de la lógica del sistema web y la API. La siguiente Figura 28 describe el modelo explicado.





**Figura 28:** *Diagrama de Despliegue*

#### 4.2.1.10 Requisitos no funcionales:

- La arquitectura propuesta por poseer diferentes componentes que la integran debe contar con una interfaz de usuario de fácil uso que logre integrar los diferentes elementos y que permita acceder a los modelos tridimensionales generados a los pacientes.
- Debe poseer un conjunto de subrutinas, métodos o funciones que logren interactuar con el prototipo del dispositivo móvil y éste a su vez realice el despliegue tridimensional del estudio del paciente de forma portable dada a la cantidad de dispositivos que existen en el mercado.
- Tanto el sistema web como el prototipo móvil se deben acceder de manera segura ya que poseen información sensible que no deben ser observada por personal no autorizado.

- La arquitectura debe ofrecer resultados confiables ya que se maneja información sensible de los pacientes.
- El prototipo móvil debe ser escalable para manejar un crecimiento continuo de trabajo debido a la recurrencia de pacientes que pueden llegar al centro asistencial.
- Se debe contar con alto espacio de almacenamiento y buen rendimiento a la hora de acceder a los datos según el volumen de pacientes y estudios que se maneje.

#### **4.2.1.11 Especificaciones:**

Es necesario que la arquitectura considere ciertos detalles necesarios para su implementación en la plataforma dada, acompañada de los procesos y estructuras que la componen. Entre ellos existen algunas características técnicas y requisitos mínimos que logren el correcto funcionamiento del prototipo móvil y tanto del sistema web como el cliente que lo accede.

##### ***a. Características Técnicas Sistema Web:***

Los requisitos mínimos básicos de software para establecer la arquitectura propuesta basada en software libre se pueden dividir en distintas áreas, el cliente web, el servidor web, el repositorio de modelos, el servidor de base de datos y el dispositivo móvil. Dada la versatilidad de la arquitectura, el servidor web, el repositorio de modelos y el servidor de base de datos se pueden integrar en un solo servidor, presentando los siguientes requisitos de software:

- Apache 2.4.10
- PHP 5.6.3
- Sistema Operativo Ubuntu 14.04.2 LTS para los servidores

Los requisitos mínimos para el cliente web son los comunes para una estación de trabajo, el cual necesita de lo siguiente:

- Navegador Mozilla Firefox 36.0.4 para consular el sistema web
- Sistema Operativo Multiplataforma (Windows, Linux, Mac)

***b. Requisitos de hardware sistema web:***

Para una institución con mediana afluencia de pacientes y para mantener un nivel de desempeño óptimo entre 1 a 10 conexiones simultáneas, y conservar el mismo enfoque del servidor web, el repositorio de modelos y el servidor de base de datos en un solo servidor, se requiere de lo siguiente:

- 1 TB Disco Duro para el servidor
- 4GB Memoria para el servidor
- Tarjeta de red Ethernet 10/100mbps para la conexión a la arquitectura
- Conexión de Red Inalámbrica
- Teclado
- Monitor

Los requisitos mínimos para el cliente web son los comunes para una estación de trabajo, el cual necesita de lo siguiente:

- 500 GB Disco Duro
- Tarjeta de red Ethernet 10/100mbps para la conexión a la arquitectura
- 1GB Memoria RAM
- Teclado
- Monitor
- Ratón

***c. Características Técnicas prototipo móvil:***

El requisito mínimo de software para ejecutar el prototipo propuesto en un dispositivo móvil es Android 4.2.2.

***d. Requisitos de hardware prototipo móvil:***

Los requisitos mínimos básicos de hardware para ejecutar el prototipo propuesto son los de un dispositivo móvil de alta gama, el cual son los siguientes:

- 2 GB de RAM
- 300 MB de espacio libre
- CPU Quad Core 1.5Ghz
- Wi-Fi 802.11 b/g/n
- Pantalla de 4.5 pulgadas

*e. Requisitos de almacenamiento:*

Entre los requisitos de almacenamiento, el tipo de almacenamiento, la ubicación de almacenamiento de la base de datos y de los archivos de registro, el sistema web que forma parte de los componentes de la arquitectura se usa como gestor de base de datos Mysql 5.6.21.

*f. Diseño de la base de datos para el sistema web:*

Todos los datos son almacenados en una base de datos centralizada que permite acceder a los registros de los modelos tridimensionales para ser descargados y desplegados en el dispositivo móvil a través del prototipo propuesto bajo la arquitectura dada. Los registros son cargados a sus respectivas tablas mediante el sistema web, como se ha mencionado, cada opción del menú principal corresponde a una tabla creada para un propósito específico dentro del sistema.

La base de datos consta de 10 tablas que permiten el uso básico del sistema y prototipo, entre ella se encuentra el País (Almacena el País de procedencia de la persona), Estado Civil (Registra el estado civil de la persona), Género (Guarda el sexo de la persona), Persona (Representa todos los usuarios, pacientes o médicos del sistema), Usuario (Registra todas aquellas personas con credenciales para ingresar al prototipo propuesto), Paciente (todas aquellas personas a los cuales se les hace un examen médico a través de una tomografía u otro tipo de estudio en modelo 3D), Médico (Personal especialista encargado de evaluar los exámenes de los pacientes registrados en el sistema), Tipo de Especialidad (conjunto de conocimientos médicos especializados de los médicos), Examen Tomografía (Cada uno de los estudios generados a los pacientes, según el

médico para ser desplegados en el dispositivo móvil) y Tipo de Examen que es el tipo de estudio generado.

A continuación se muestra el diagrama Entidad – Relación (Figura 29) de la base de datos para tener un mayor detalle de los campos, tipos de datos, longitud, claves primarias, claves foráneas, campos únicos y relaciones entre cada una de las tablas anteriormente expuestas:

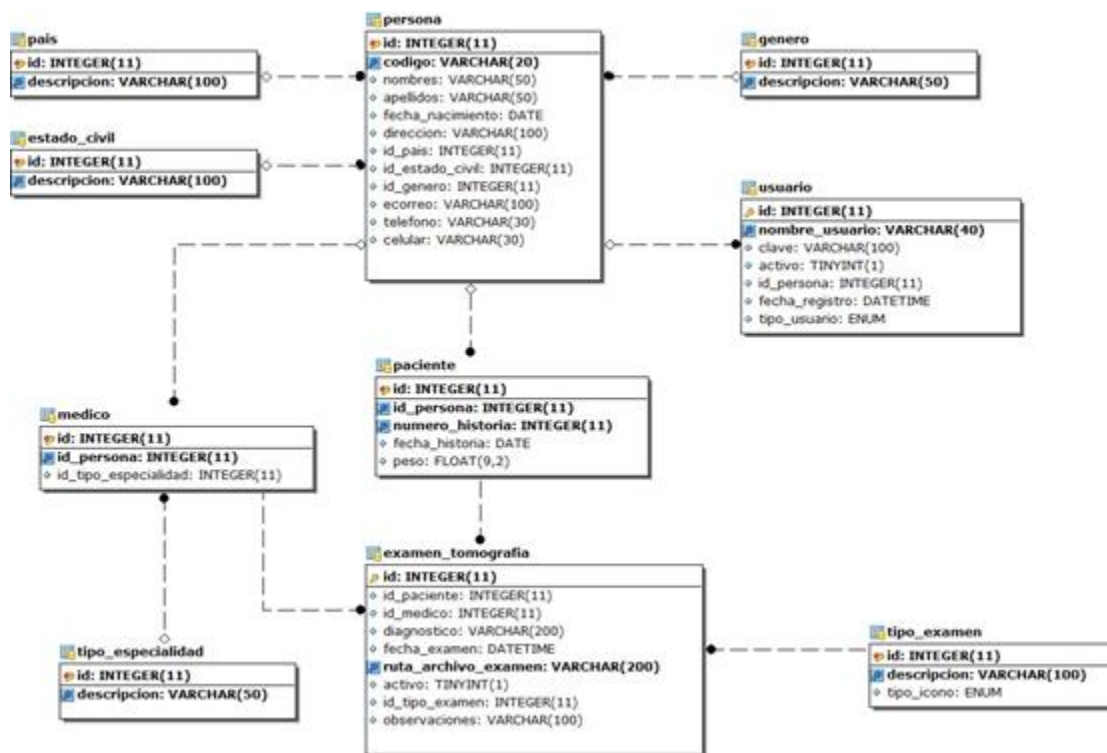


Figura 29: Modelo entidad relación Base de datos propuesta

## CAPÍTULO V

### IMPLEMENTACIÓN Y FUNCIONAMIENTO DE LA ARQUITECTURA PROPUESTA

En el presente capítulo se hace una descripción de cada uno de los elementos que integran la plataforma propuesta, entre ellos el repositorio de modelos, la API, el cliente web, el sistema web, la estructura de la API empleada y el prototipo del dispositivo móvil

#### 5.1 Arquitectura Propuesta

La arquitectura propuesta en el presente proyecto de investigación se puede observar en la Figura 30, el cual, comienza por la captura de las imágenes médicas que pueden ser obtenidas de diferentes fuentes, como la Tomografía computarizada, ultrasonido, resonancia magnética, entre otros. Estas imágenes se encuentran en formato DICOM y se envían al Sistema de Archivo y Comunicación de Imagen (PACS) para su posterior procesamiento tridimensional según las regiones de interés y el tipo de estudio a efectuar por parte de los especialistas.

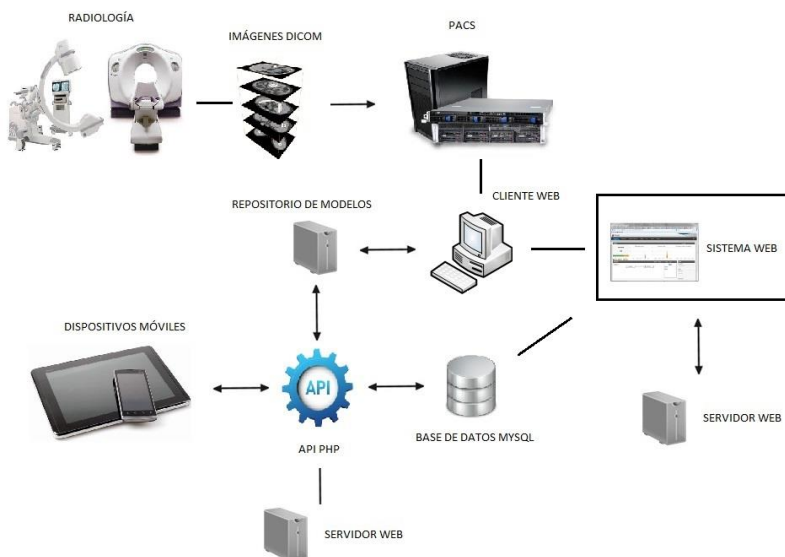
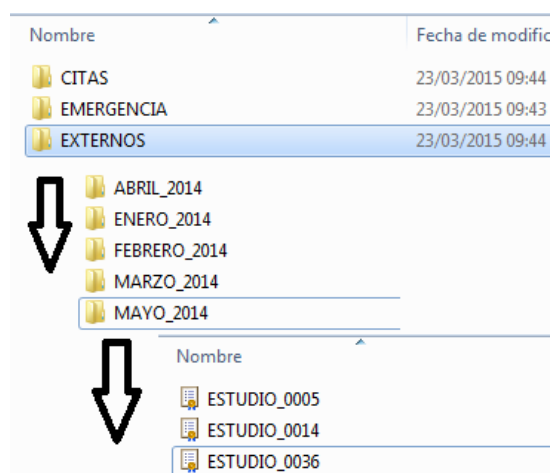


Figura 30: Arquitectura propuesta

## 5.2 Repositorio de modelos

Un repositorio puede estar organizado para todo el personal encargado de generar el estudio; puede ser dividido en carpetas para diferentes departamentos o áreas de trabajo o para un usuario individual, en otras palabras, la organización interna del repositorio de archivos es dada según las necesidades de la organización donde se encuentre la arquitectura propuesta. En la Figura 31 se muestra un ejemplo en la forma como se pudiera organizar el repositorio, allí se puede observar la forma en que se encuentran divididos los estudios, ya sea todos aquellos pacientes que entraron por citas, por emergencia, o cualquier otro examen externo a la organización. Dentro de ellos se localiza otra división según el mes y año del estudio y dentro de estos últimos están los modelos en formato tridimensionales con un nombre dado a conveniencia.



**Figura 31:** Organización del repositorio de modelos 3D

Continuando con el repositorio de modelos es importante mencionar que la gestión del directorio debe ser hecha por un administrador del sistema que tenga permisos de escritura para los directorios del servidor, en donde se debe llevar a cabo el servicio de administración de usuarios y permisología pertinente para reflejar la seguridad requerida por la dependencia que solicita el servicio. El repositorio de modelos es multiplataforma, por lo que se puede implantar en cualquier sistema operativo con conexión a la red para la carga y descarga de los estudios.

### **5.3 Cliente web**

Una vez generado el estudio tridimensional del área de interés del paciente el personal encargado se conecta a un sistema web desarrollado en PHP a través de un navegador para ingresar los resultados del examen junto con el modelo 3D (ruta en el repositorio), paciente atendido, médico encargado, resultados observados, fecha del examen, entre otros datos por medio de un servidor web que sirve como plataforma para realizar las conexiones bidireccionales con el cliente mediante el protocolo HTTP. A su vez el sistema web se conecta a la base de datos centralizada Mysql donde se encuentra la información de la ruta del modelo 3D junto con los datos de los pacientes, médicos y estudios efectuados para cada uno de ellos.

### **5.4 Sistema web**

El cliente web debe tener una conexión a la red para poder acceder al sistema web desarrollado en PHP en la dirección donde se encuentre alojado el servidor, pudiera ser en la nube, VPN o en la Red de Área Local dependiendo de las características y necesidades de la organización. El sistema web se encarga de alimentar, modificar o consultar la base de datos con el contenido requerido para generar el despliegue tridimensional. Al ingresar al URL, se muestra una página para introducir las credenciales para ingresar al sistema, tal y como se muestra en la Figura 32. Allí es requerido un nombre de usuario y contraseña para acceder al sistema encargado de ingresar, modificar, eliminar y visualizar la información respectiva a los modelos tridimensionales que se despliegan en el dispositivo móvil almacenados en la base de datos y repositorio de modelos descritos anteriormente.



**Figura 32:** Inicio de sesión del sistema web

En dicho sistema es posible gestionar el estado civil de las personas registradas, el examen de la tomografía, el género de las personas, los médicos, pacientes, países para identificar las nacionalidades, las personas, el tipo de especialidad para los médicos, el tipo de examen o estudio y los usuarios del sistema. En la siguiente Figura 33 en la parte izquierda se puede observar una captura de pantalla del navegador con el menú principal de las opciones anteriormente descritas. Al seleccionar cada una de ellas se muestra un listado con el contenido almacenado en la base de datos en donde se puede efectuar cada una de las operaciones básicas como se muestra a continuación:

Acciones				Codigo
Ver	Editar	Eliminar	Copiar	2
Ver	Editar	Eliminar	Copiar	4
Ver	Editar	Eliminar	Copiar	1
Ver	Editar	Eliminar	Copiar	3

**Figura 33:** Menú principal sistema web

Una vista más detallada se puede observar en la Figura 34, en este caso para el tipo de especialidad de los médicos. Como se indicó anteriormente, para cada una de las opciones del menú principal, se genera un listado con su información respectiva. Adicionalmente se despliega una barra de herramientas que como se observa, permite Agregar, Actualizar el listado, efectuar una búsqueda rápida sobre todas las columnas y las Acciones sobre cada uno de los registros.

### Tipo Especialidad

Acciones				Codigo <sup>▲</sup>	Descripcion
Ver	Editar	Eliminar	Copiar	1	CIRUJANO
Ver	Editar	Eliminar	Copiar	2	TRAUMATOLOGO
Ver	Editar	Eliminar	Copiar	3	UROLOGO
Ver	Editar	Eliminar	Copiar	4	NEUROCIRUJANO
Ver	Editar	Eliminar	Copiar	5	OBSTETRA

**Figura 34:** Menú herramientas sistema web

Uno de los formularios más importantes y utilizados es el examen de tomografía, el cual se indica el paciente a quien se le generó el estudio, el médico responsable, el diagnóstico, la fecha del examen, la ruta del archivo donde se encuentra el estudio en el repositorio de modelos, si el examen se encuentra activo o no, el tipo de examen y algunas observaciones por cada fila en la base de datos. En la Figura 35 se muestra el formulario utilizado en el sistema:


### Examen Tomografia

Grabar Cancelar

**Paciente \*** Por favor seleccione... ▾

**Medico \*** Por favor seleccione... ▾

**Diagnostico \***

**Fecha Examen \***  

**Ruta Archivo Examen \***

**Activo \***  No  Si

**Tipo Examen \*** Por favor seleccione... ▾

**Observaciones**

\* - Campo requerido

Grabar Cancelar

**Figura 35:** *Formulario carga del examen en sistema web*

Es posible registrar todos los usuarios del prototipo para su posterior uso por parte de los médicos. Es requerido introducir el nombre del usuario el cual debe ser único que no se puede repetir, la contraseña sin restricciones de caracteres o tamaño, estado del usuario (Activo o Inactivo), la persona asociada al usuario, la fecha de registro y el tipo de usuario (Usuario o Administrador). Las contraseñas son cifradas en MD5 concatenando la propia clave junto con la fecha de registro para proporcionar mayor seguridad a los mismos usuarios que se genera por medio de un trigger almacenado en la tabla de usuario que ese activa antes de actualizar y antes de insertar cada registro. En la Figura 36 se encuentra el formulario anteriormente descrito.

## Usuario

Nombre Usuario \*

Clave \*

Activo \* No

Persona \* Por favor seleccione...

Fecha Registro \*

Tipo Usuario \* Por favor seleccione...

\* - Campo requerido

**Figura 36:** *Formulario registro de usuarios en sistema web*

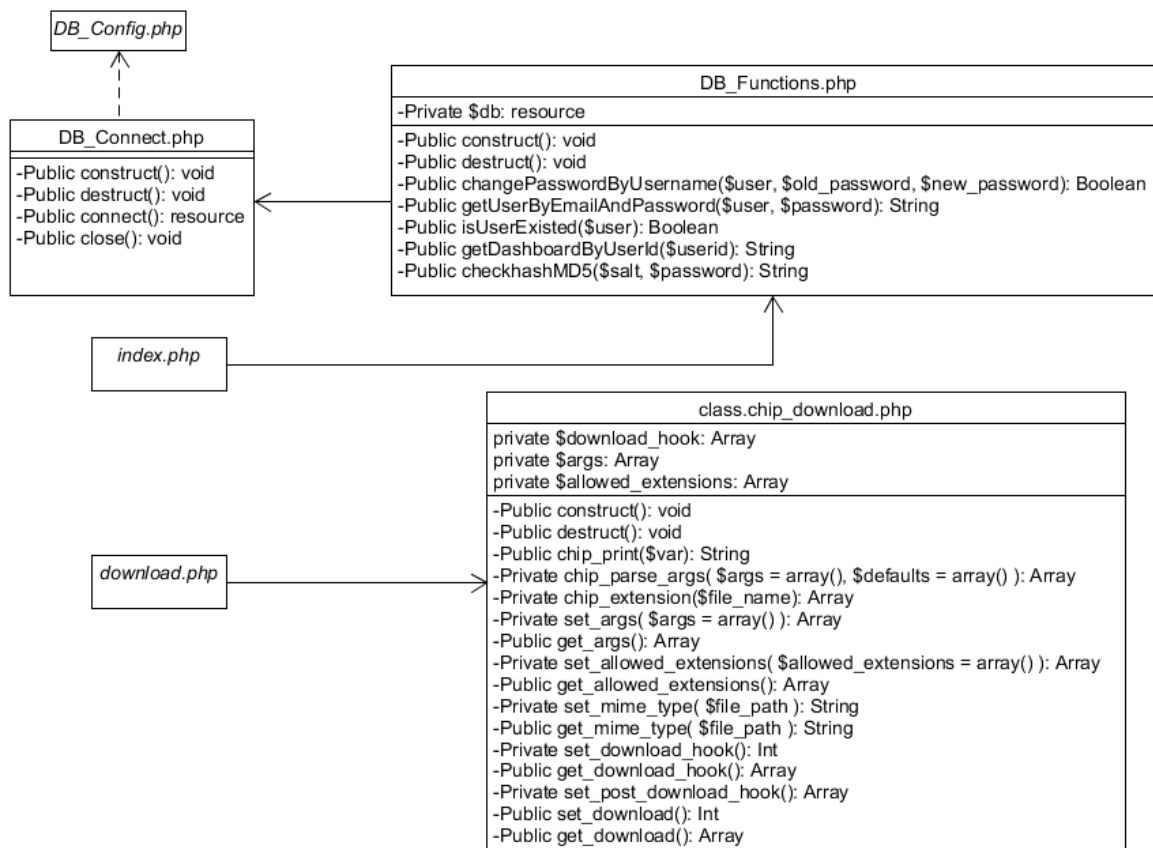
## 5.5 API PHP

La interfaz de programación de aplicaciones (API) contiene el conjunto de métodos que permiten ser utilizados para interactuar entre el prototipo y la base de datos y el repositorio de modelos dentro del entorno de la red. Específicamente consta de cuatro tipos de solicitudes, entre ellas, para realizar el inicio de sesión al prototipo, consultar la Bandeja de Exámenes, modificar la contraseña del usuario y cerrar la sesión utilizando el lenguaje de programación PHP. Cada tipo de solicitud es identificado por un parámetro que es capturado por la API mediante el método POST enviado por el usuario a través del prototipo para que posteriormente la API responda de nuevo al prototipo. Al igual que el sistema web, la API PHP se encuentra alojada en el servidor web, que como se mencionó anteriormente, se encarga de gestionar la conexión de la misma API PHP con el resto de componentes de la arquitectura.

### 5.5.1 Estructura API PHP

La estructura y clases de la API se puede observar en la Figura 37, que se encuentra conformada por seis archivos en PHP cada uno de ellos cumple una función específica:

- **DB\_Config.php:** Contiene la dirección, usuario, contraseña y nombre de la base de datos.
- **DB\_Connect.php:** Abre y cierra la conexión a la base de datos.
- **DB\_Functions.php:** Contiene los métodos para interactuar con la base de datos, entre ellos cambiar la contraseña, obtener usuario por nombre y clave, determinar si el usuario existe en la base de datos y obtener la bandeja de exámenes.
- **Index.php:** Maneja todas las solicitudes HTTP del usuario y las respuestas al prototipo en formato JSON. Cada solicitud es identificada por una etiqueta que es capturada mediante el método POST de PHP. El valor de la etiqueta debe ser *login*, *dashboard*, *change\_password* y *logout*, que es para el inicio de sesión, bandeja de exámenes, cambiar contraseña y cierre de sesión respectivamente.
- **Download.php:** Maneja todas las solicitudes de descarga de los modelos 3D almacenados en el repositorio. Recibe como parámetro el nombre del archivo .stl
- **Class.Chip\_download.php:** Contiene los métodos requeridos para descargar los modelos 3D.



**Figura 37:** Diagrama estructura API PHP

## 5.5.2 Solicitud y Respuesta API PHP

Como se mencionó, entre el dispositivo móvil y la API PHP existen distintos tipos de solicitudes y respuestas. Las solicitudes contienen los valores de los formularios en las pantallas del prototipo y la respuesta está dada en formato JSON (*JavaScript Object Notation*, que significa Notación de Objetos de JavaScript) para todos los casos menos el proceso de cerrar sesión. Cada uno de los tipos de solicitudes se describe a continuación:

### 5.5.2.1 Inicio de Sesión

Permite validar las credenciales del usuario registrado en la base de datos, la entrada y salida se describe en la tabla 23.

<b>NOMBRE DEL PROCESO:</b>	Inicio de Sesión
<b>VALORES DE ENTRADA (POST):</b>	Nombre de Usuario (String) Contraseña (String) Tag ( <i>login</i> ) (Constante)
<b>SALIDA:</b>	Respuesta JSON (String)

**Tabla 7:** Entrada/Salida proceso inicio de sesión.

La salida puede variar según el inicio de sesión fue correcto o incorrecto, la siguiente tabla 24 describe el formato JSON para ambos casos:

<b>Definición Inicio Sesión Correcto</b>	<b>Ejemplo</b>
<pre>{   "tag": &lt;Nombre del parámetro&gt;,   "success": &lt;Inicio Exitoso o no&gt;,   "error": &lt;Error o no&gt;,   "id": &lt;Código del usuario&gt;,   "user": {     "name": &lt;Nombre del usuario&gt;,     "created_at": &lt;Fecha de registro&gt;   } }</pre>	<pre>{   "tag": "login",   "success": 1,   "error": 0,   "id": "23",   "user": {     "name": "carlos",     "created_at": "2015-03-01 13:27:05"   } }</pre>
<b>Definición Inicio Sesión Incorrecto</b>	<b>Ejemplo</b>
<pre>{   "tag": &lt;Nombre del parámetro&gt;,   "success": &lt;Inicio Exitoso o no&gt;,   "error": &lt;Código del error&gt;,   "error_msg": &lt;Mensaje de error&gt;</pre>	<pre>{   "tag": "login",   "success": 0,   "error": 1,   "error_msg": "Usuario o contraseña"</pre>

}	incorrecta"
	}

**Tabla 8:** Definición inicio de sesión correcta o incorrecta en JSON.

### 5.5.2.2 Consultar Bandeja de Exámenes

Permite consultar todos los exámenes del médico (usuario) respecto a sus pacientes, la entrada y salida se describe en la tabla 25:

<b>NOMBRE DEL PROCESO:</b>	Consultar Bandeja de Exámenes
<b>VALORES DE ENTRADA (POST):</b>	Código del usuario (String) Tag ( <i>dashboard</i> ) (Constante)
<b>SALIDA:</b>	Respuesta JSON (Array)

**Tabla 9:** Entrada/Salida proceso Consultar Bandeja de Exámenes.

La salida puede variar según el usuario posee exámenes de pacientes en la bandeja de exámenes o no, en el caso que un médico no posea exámenes activos. La tabla 26 describe el formato JSON para ambos casos:

Definición Consultar Bandeja de Exámenes con data	Ejemplo
<pre>[{   "idExamen": &lt;Código Examen&gt;,   "especialidadMedico": &lt;Especialidad Médico&gt;,   "codigoMedico": &lt;Código Médico&gt;,   " nombreMedico ": &lt;Nombre Médico&gt;,   "codigoPaciente": &lt;Código Paciente&gt;   "nombrePaciente": &lt;Nombre Paciente&gt;,</pre>	<pre>[{"0":"16","idExamen":"16","1":"TRAUMATOLOGO","especialidadMedico":"TRAUMATOLOGO","2":"V12576544","codigoMedico":"V12576544","3":"JAVIER REYES","nombreMedico":"JAVIER REYES","4":"V1235512","codigoPacie</pre>



<pre>"numero_historia": &lt;Número Historia&gt;, "fechaHistoria": &lt;Fecha Historia&gt;, "fechaExamen": &lt;Fecha Examen&gt;, "rutaExame": &lt;Ruta Examen&gt;, "observaciones":&lt;Observaciones&gt;, "idUsuario": &lt;Código Usuario&gt;, "tipoExamen": &lt;Tipo Examen&gt;, "tipoIcono": &lt;Tipo Icono&gt; }, {...}]</pre>	<pre>nte":"V1235512","5":"ANDREA MARTINEZ","nombrePaciente":"AND REA MARTINEZ","6":"1244","numero_hist oria":"1244","7":"2014-12- 03","fechaHistoria":"2014-12- 03","8":"02\03\2015      10:34:00 AM","fechaExamen":"02\03\2015 10:34:00 AM","9":"Pelvis2.stl","rutaExamen":"P elvis2.stl","10":"qwert","observaciones" :"qwert","11":"1","idUsuario":"1","12": "TOMOGRAFIA","tipoExamen":"TOM OGRAFIA","13":"type3","tipoIcono":"t ype3"}, {...}]</pre>
<p align="center"><b>Definición Consultar Bandeja de Exámenes sin data</b></p>	<p align="center"><b>Ejemplo</b></p>
<pre>{ "tag": &lt;Nombre del parámetro&gt;, "success": &lt;Inicio Exitoso o no&gt;, "error": &lt;Código del error&gt;, "error_msg": &lt;Mensaje de error&gt; }</pre>	<pre>{ "tag": "dashboard", "success": 0, "error": 3, "error_msg": "Data no encontrada" }</pre>

**Tabla 10:** Definición inicio de sesión correcta o incorrecta en JSON.

### 5.5.2.3 Cambiar Contraseña

Permite modificar la clave del usuario registrado en la base de datos, la entrada y salida se describe en la tabla 27:

<b>NOMBRE DEL PROCESO:</b>	Cambiar Contraseña
<b>VALORES DE ENTRADA (POST):</b>	Nombre de Usuario (String) Contraseña anterior (String) Contraseña Nueva (String) Tag <i>(change_password)</i> (Constante)
<b>SALIDA:</b>	Respuesta JSON (String)

**Tabla 11:** *Entrada/Salida proceso cambiar contraseña.*

La salida puede variar según el inicio de sesión fue correcto o incorrecto, la siguiente tabla 28 describe el formato JSON para ambos casos:

<b>Definición Cambio de Clave Correcto</b>	<b>Ejemplo</b>
<pre>{ "tag": &lt;Nombre del parámetro&gt;, "success": &lt;Inicio Exitoso o no&gt;, "error": &lt;Error o no&gt;, "id": &lt;Código del usuario&gt;, "user": { "name": &lt;Nombre del usuario&gt;, "created_at": &lt;Fecha de registro&gt; } }</pre>	<pre>{ "tag": " change_password ", "success": 1, "error": 0, "id": "23", "user": { "name": "carlos", "created_at": "2015-03-01 13:27:05" } }</pre>
<b>Definición Cambio de Clave Incorrecto</b>	<b>Ejemplo</b>
<pre>{ "tag": &lt;Nombre del parámetro&gt;, "success": &lt;Inicio Exitoso o no&gt;, "error": &lt;Código del error&gt;,</pre>	<pre>{ "tag": " change_password ", "success": 0, "error": 2,</pre>

"error_msg": <Mensaje de error> }	"error_msg": "Usuario no existe" }
--------------------------------------	---------------------------------------

**Tabla 12:** Definición inicio de sesión correcta o incorrecta en JSON.

#### 5.5.2.4 Cerrar de Sesión

Permite cerrar la sesión del usuario, es decir, hacer se invoca al método *session\_destroy()* de PHP. La entrada y salida se describe en la siguiente tabla 29:

<b>NOMBRE DEL PROCESO:</b>	Cerrar sesión
<b>VALORES DE ENTRADA (POST):</b>	Código de Usuario (String) Tag ( <i>logout</i> ) (Constante)
<b>SALIDA:</b>	No posee

**Tabla 13:** Entrada/Salida proceso cerrar sesión.

## 5.6 Prototipo de Dispositivo Móvil

El desarrollo del prototipo permite realizar un despliegue tridimensional sobre los modelos en formato .STL (STereo Lithography) sobre la arquitectura dada, permitiendo al personal especializado hacer un diagnóstico sobre la región de interés del paciente de manera digital en el dispositivo móvil conectado a la red. Para ello se ha desarrollado un acceso básico sobre las operaciones que debe realizar el médico en el celular o tabla.

### 5.6.1 Entorno de software

Para logra los objetivos planteados se hizo uso de herramientas de cómputo para la elaboración de aplicaciones en Android, particularmente el entorno de desarrollo para aplicaciones Android Studio 1.0 que es gratuito bajo la licencia Apache 2.0 en un entorno Windows 7 Home Basic y que utiliza como lenguaje de programación orientado a objetos Java. Adicionalmente se tomó la aplicación denominada *STL Viewer for Android* como librería para la

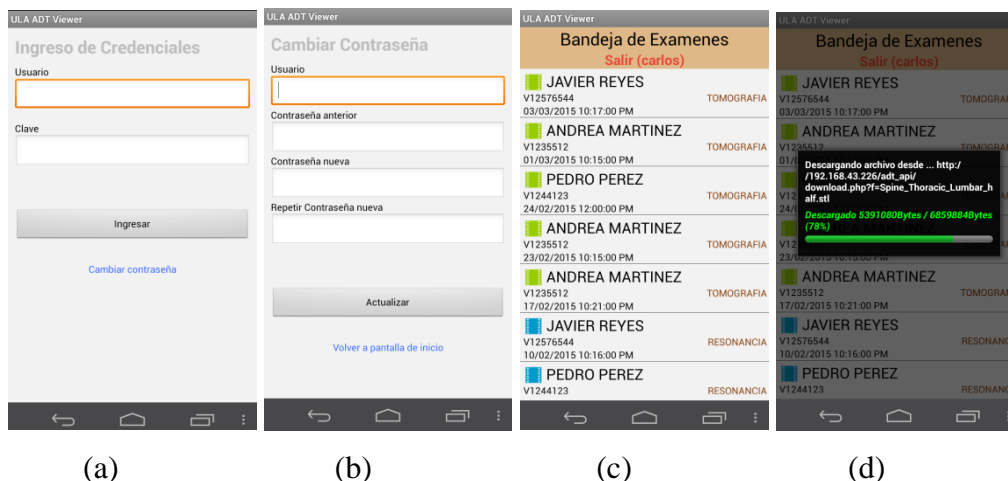
visualización de objetos 3D que utiliza OpenGL como estándar de API gráfica multiplataforma para el procesamiento tridimensional.

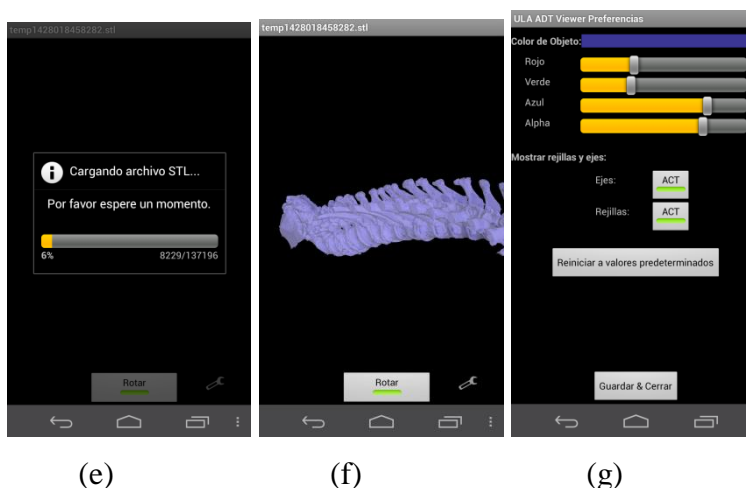
### 5.6.2 Funcionamiento del prototipo

Para entender de mejor manera el funcionamiento del prototipo en el dispositivo móvil, se muestran el siguiente diagrama de Actividades. Para comenzar, el usuario inicia la sesión, de ser requerido puede modificar su contraseña, al hacerlo, vuelve a la pantalla de inicio de sesión. Luego que el prototipo valida las credenciales, se carga la bandeja de exámenes donde el mismo usuario selecciona el estudio del paciente deseado para luego hacer el análisis o diagnóstico del modelo tridimensional, de ser requerido el usuario puede modificar las preferencias de visualización como el color o nivel de transparencia del objeto. Al culminar, el usuario puede volver a la bandeja de entrada y seleccionar otro examen o cerrar la sesión y salir.

### 5.6.3 Pantallas y formularios del prototipo

La Figura 38 muestra cada una de las pantallas y/o formularios que contiene el prototipo, de manera general, en la Figura 38 (a) se muestra el formulario de ingreso de credenciales, con los campos de texto del nombre de usuario y clave, el botón de ingresar y de cambiar contraseña. En la Figura 38 (b) se muestra el formulario para cambiar la contraseña del usuario, con los campos de texto de usuario, contraseña anterior, contraseña nueva y repetir contraseña nueva junto con los botones para actualizar y para volver a la pantalla de inicio.





**Figura 38:** Vista del Prototipo. (a) Inicio de Sesión. (b) Cambio de contraseña. (c) Bandeja de Exámenes. (d) Descarga de modelo 3D. (e) Carga de archivo STL. (f) Despliegue 3D del modelo. (g) Preferencias Despliegue 3D

En la Figura 38 (c) se encuentran en la pantalla de Bandeja de Exámenes con el listado (nombre y cédula del paciente, fecha del estudio y tipo de estudio) de todos los estudios activos de los pacientes y el botón de salir para cerrar la sesión. Al presionar cualquier estudio de dicho listado se despliega la interfaz de la Figura 38 (d) con la barra de progreso de descarga del archivo 3D. Al finalizar la descarga, se activa la pantalla de la Figura 38 (e) con una nueva barra de progreso de carga de cada una de las líneas del archivo STL. Al culminar, se muestra el propio modelo tridimensional que se encuentra en la Figura 38 (f) el cual el usuario puede rotar o enfocar a través de la pantalla táctil del dispositivo. Allí mismo se encuentra un icono que permite modificar las preferencias de despliegue, que se observa en la Figura (f) permitiendo modificar la combinación de colores RGB (Rojo, Verde y Azul), el nivel de transparencia (alpha), activar o desactivar los ejes y rejillas en la escena del modelo y reiniciar los valores predeterminados.

#### 5.6.4 Configuración para el funcionamiento del prototipo en la red de la arquitectura

Para poner en funcionamiento el prototipo dentro de la arquitectura en un entorno de red es necesario modificar una serie de parámetros en los archivos del código fuente. Para comenzar, dentro del paquete *Library* (librería) del código fuente del prototipo se encuentra la clase

*ServerSetting* (Código de Programación 1), allí se conFiguran los atributos que permiten indicar la dirección donde se encuentra la API y la dirección donde se encuentra el archivo PHP para descargar los modelos, además el nombre de los parámetros *tag* correspondiente a cada llamada a la api en php

```

/* ConFiguración de dirección de la api
 * @author Carlos
 * @version 1
 */
public final class ServerSetting {
    public static String api_url = "http://192.168.43.226/adt_api/";
    public static String login_tag = "login";
    public static String change_password_tag = "change_password";
    public static String dashboard_tag = "dashboard";
    public static String logout_tag = "logout";
    public static String download_file_path =
"http://192.168.43.226/adt_api/download.php?f=";
}

```

### ***Código de Programación 1: Clase ServerSetting***

Dentro de la API se encuentra el archivo llamado *DB\_config.php* el cual permite modificar los parámetros de conexión a la base de datos, tal y como se muestra a continuación.

```

<?php
/**
 * Variables de conFiguración de la base de datos
 */
define("DB_HOST", "localhost");
define("DB_USER", "root");
define("DB_PASSWORD", "");
define("DB_DATABASE", "adt_bd");
?>

```

### ***Código de Programación 2: Archivo DB\_config.php***

Por último, en el archivo `download.php` dentro de la API, en la línea 27 se define el parámetro *FSROOT* el cual apunta a la dirección donde se encuentra el repositorio de modelos 3D.

## 5.7 Renderización de la escena 3D

Para el proceso de despliegue tridimensional es necesario representar el modelo en la pantalla del dispositivo móvil. Dicho modelo se encuentra en formato STL descrito en base a triángulos, sin información de color, texturas, capas, tipos de trazo, superficie o propiedades físicas, por lo tanto, es una representación triangular de una geometría de la superficie de 3 dimensiones que se descompone lógicamente en una serie de triángulos pequeños o caras descrita por una dirección perpendicular y tres puntos que representan los vértices del triángulo. Estos datos son utilizados por un algoritmo de corte para determinar las secciones transversales de la forma 3-dimensional.

Para generar la escena 3D a partir del modelo STL se utiliza la librería *STLViewer* desarrollada por Shoji (2011) bajo licencia Apache versión 2.0 el cual utiliza la API OpenGL para dibujar los puntos, líneas, triángulos, iluminación y color que facilita el diagnóstico por parte del especialista. Como se menciona en la fase de diseño, el prototipo creado cuenta con el paquete *Renderer* que a su vez posee la clase *STLRenderer* que se encarga de dibujar cada línea y hacer los movimientos de traslación, rotación, acercamiento y alejamiento para los objetos de tipo *STLObject* contenidos dentro del paquete *object* el cual se encarga de convertir el archivo en formato STL a un objeto con las posiciones X, Y y Z de cada vértice contenido.

El proceso de despliegue comienza leyendo el archivo en formato STL el cual se describe en el código de programación 3 que es un fragmento del código fuente contenido en la clase *STLObject.java*. Allí se muestra como lee por cada línea de dicho archivo cada una de las caras de los triángulos y sus respectivos vértices en un vector de caras llamado *normalList* y un vector de vértices llamado *vertexList*. Para mejorar la escena se usa el método llamado *djustMaxMin*

para justar los valores máximos y mínimos del objeto en la pantalla para mostrarse posteriormente en la pantalla del dispositivo.

```

for (int i = 0; i < stlLines.length; i++) {
    String string = stlLines[i].trim();
    if (string.startsWith("facet normal ")) {
        string = string.replaceFirst("facet normal ", "");
        String[] normalValue = string.split(" ");
        normalList.add(Float.parseFloat(normalValue[0]));
        normalList.add(Float.parseFloat(normalValue[1]));
        normalList.add(Float.parseFloat(normalValue[2]));    }

    if (string.startsWith("vertex ")) {
        string = string.replaceFirst("vertex ", "");
        String[] vertexValue = string.split(" ");
        float x = Float.parseFloat(vertexValue[0]);
        float y = Float.parseFloat(vertexValue[1]);
        float z = Float.parseFloat(vertexValue[2]);
        djustMaxMin(x, y, z);
        vertexList.add(x);
        vertexList.add(y);
        vertexList.add(z); }

```

**Código de Programación 3. Lectura STL ASCII (STLObject.java)**



## CAPÍTULO VI

### PRUEBAS FUNCIONALES Y DE RENDIMIENTO

El presente capítulo hace énfasis en las pruebas de funcionamiento y rendimiento para asegurar el correcto desenvolvimiento de los componentes de software programados y la calidad de los mismos respectivamente.

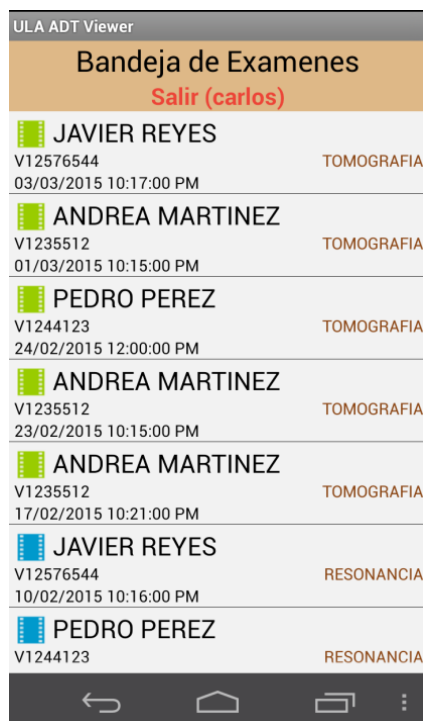
#### 6.1 Pruebas funcionales

En este tipo de prueba se busca revisar el funcionamiento y comportamiento correcto para cada módulo del sistema. En la tabla 30 se muestra la prueba de funcionamiento para el listado de la bandeja de exámenes del médico especialista en el dispositivo móvil.

<b>Nombre Caso de Prueba</b>	Listar exámenes médicos
<b>Descripción</b>	El Módulo busca en la base de datos del sistema web los estudios de los pacientes que posee el médico previamente cargados.
<b>Tipo</b>	Prueba de Unidad
<b>Entrada</b>	Código del usuario del médico
<b>Procedimiento</b>	Consultar en la base de datos la información del paciente, tipo de examen y fecha Mostrar en la pantalla los resultados obtenidos
<b>Resultado esperado</b>	Listado de todos exámenes de los pacientes tratados por el médico.
<b>Resultados obtenidos</b>	Se genera un resultado con nombre de los pacientes, cedula, fecha y hora del estudio, tipo de estudio, y un ícono de diferente color según el mismo.

**Tabla 14:** Cuadro de prueba de funcionamiento para listado de estudiantes.

En la Figura 39 se puede observar cómo se hace el despliegue en la pantalla del dispositivo:



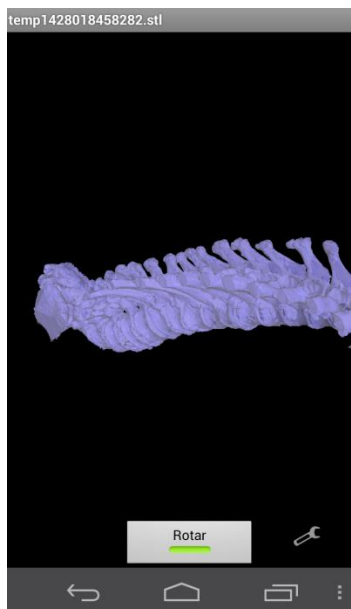
**Figura 39:** Despliegue del listado bandeja de exámenes

En la tabla 31 se muestra la prueba de funcionamiento del despliegue 3D del examen en el dispositivo móvil y en la Figura 40 como se observa dicho modelo en la pantalla del mismo.

<b>Nombre Caso de Prueba</b>	Despliegue 3D del examen médico
<b>Descripción</b>	El Módulo busca efectuar el despliegue tridimensional del estudio médico en el dispositivo móvil
<b>Tipo</b>	Prueba de Unidad
<b>Entrada</b>	Código del examen médico
<b>Procedimiento</b>	Consultar en la base de datos la información del examen médico y la ruta en el repositorio donde se encuentra el mismo Descargar en la memoria temporal del dispositivo el archivo en formato .STL Efectuar la visualización del archivo en la pantalla Mostrar el menú con las preferencias de visualización
<b>Resultado esperado</b>	Visualizar el modelo 3D en la pantalla del dispositivo móvil
<b>Resultados obtenidos</b>	Se descarga el archivo .STL con el modelo y se logra

	observar el estudio en la pantalla pudiendo el usuario mover, rotar y acercar o alejar el objeto.
--	---

**Tabla 15:** Cuadro de prueba de funcionamiento para listado de estudiantes.



**Figura 40:** Despliegue 3D examen médico

## 6.2 Pruebas de rendimiento

Las pruebas de rendimiento permiten medir la calidad del prototipo en condiciones particulares de trabajo ya que aspectos como velocidad, uso de memoria y la suavidad con la que su aplicación dibuja píxeles y realiza operaciones tiene un impacto significativo en la experiencia de los usuarios.

Según el sitio oficial de Android (Adroid Delevopers, 2015) sus aplicaciones operan en un entorno de recursos compartidos, y el rendimiento de su aplicación pueden verse afectados por la eficiencia con que interactúa con esos recursos en el sistema más grande. Las solicitudes también operan en un entorno multiproceso, compitiendo con otros procesos y que pueden causar problemas de rendimiento que son difíciles de diagnosticar.

### 6.2.1 Especificaciones red inalámbrica para prueba rendimiento

Red inalámbrica con seguridad WPA2 PSK con dirección IP por DHCP Bandas 2.4GHz.

### 6.2.2 Especificaciones del dispositivo móvil para prueba de rendimiento

La siguiente tabla 32 describe las características del dispositivo móvil donde se efectuaron las pruebas de rendimiento del prototipo bajo la arquitectura dada.

<b>Modelo</b>	Huawei Ascend P6
<b>Sistema operativo</b>	Android v 4.2.2 Jelly Bean
<b>Dimensiones</b>	132,65 x 65,5 x 6,18mm
<b>Pantalla</b>	4,7" TFT LCD Ultrafina con tecnología Magic Touch, Resolución: 1280 X 720 HD Corning Gorilla Glass 2
<b>Memoria interna</b>	ROM: 8GB, RAM: 2GB
<b>Memoria expansión</b>	MicroSD hasta 32GB
<b>Conectividad</b>	Bluetooth 3,0, Wi-Fi 802.11 b/g/n, USB 2.0 High-Speed
<b>Procesador</b>	Quad core 1.5 GHz CPU K3V2+Intel XMM6260

**Tabla 16:** *Especificaciones del dispositivo móvil de prueba (Tomado de Huawei Technologies Co, 2014)*

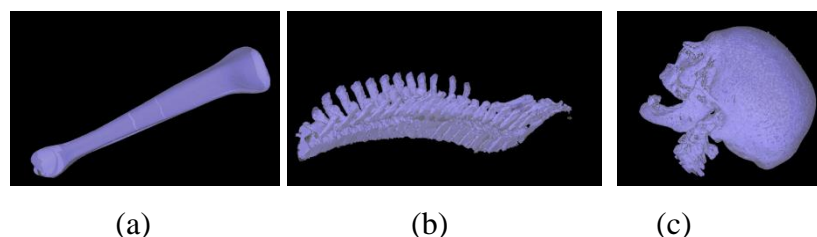
### 6.2.3 Entorno y estudios de rendimiento

El entorno de desarrollo utilizado Android Studio 1.0 posee una serie de herramientas que permiten medir el desempeño de memoria, consumo de red, uso de CPU y trazas tanto de la frecuencia de reloj del CPU como del comportamiento de una aplicación dentro del sistema operativo. En este caso se realizaron distintas pruebas con diferentes modelos de archivos en Stereolithography para determinar el rendimiento del prototipo sobre la arquitectura planteada en un entorno de desarrollo.

Los archivos utilizados para dichas pruebas son modelos que se encuentran en una comunidad web llamada Thingiverse (MakerBot Industries, LLC 2015) creada con la finalidad de descubrir, hacer y compartir objetos 3D bajo una licencia de plataforma abierta *Creative Commons*. Las características se encuentran en la siguiente tabla 33, indicando el nombre, la cantidad de caras triangulares incluidas en el vector de vértices y el tamaño en KB del archivo. En la Figura 41 se visualizan los modelos tridimensionales.

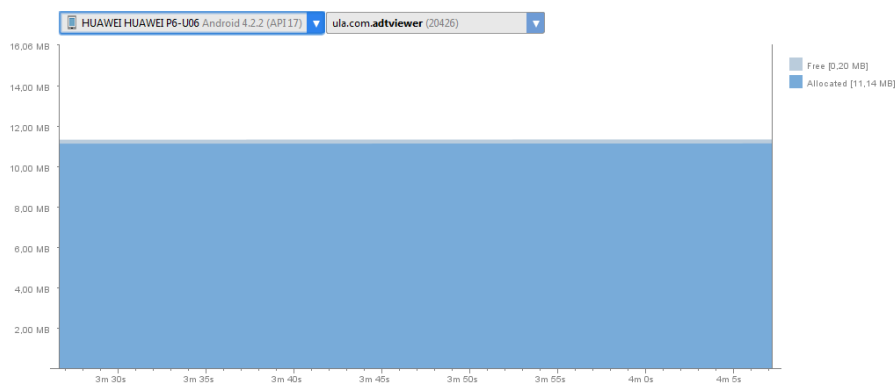
NOMBRE DEL MODELO	CARAS TRIANGULARES	TAMAÑO
Leg.stl	23.006	1.124 KB
Spine_Thoracic_Lumbar_half.stl	137.196	6.700 KB
Mandible.stl	250.276	12.221 KB

**Tabla 17:** Características de los modelos de prueba.



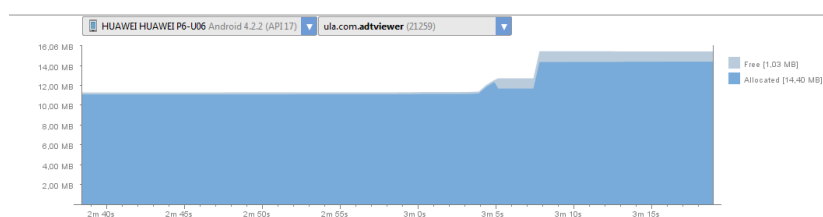
**Figura 41:** Visualización de modelos de prueba. (a) Modelo Leg.stl (b) Modelo Spine\_Thoracic\_Lumbar\_half.stl (c) Modelo Mandible.stl

Comenzando con las pruebas de rendimiento en la siguiente Figura 42 se puede observar el monitor de memoria para el consumo de la misma al transcurrir cierto tiempo sobre la pantalla de bienvenida del prototipo. Se logra identificar que a lo largo del tiempo (Eje X) se mantiene el consumo de 11.14 MB (Eje Y) sobre la gestión de memoria dada por el sistema operativo del dispositivo de prueba.



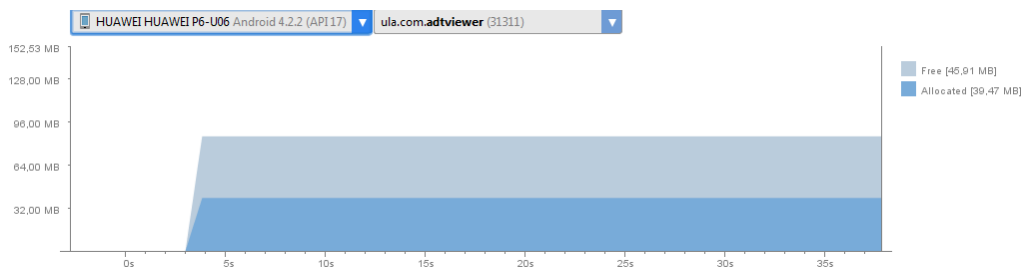
**Figura 42:** Consumo de memoria pantalla inicio prototipo

Posteriormente al ingresar exitosamente las credenciales del usuario, se lleva a la pantalla de Bandeja de Exámenes con todos los modelos para el estudio médico determinado cargados para cada uno de los pacientes del médico especialista. En este caso de prueba se cargaron un total de 20 exámenes, logrando visualizar en la siguiente Figura 43 un leve cambio sobre el consumo de memoria del prototipo, aumentando a 14.40 MB, lo que es totalmente comprensible ya que se posee mayor cantidad de información alojada en el sistema y permitiendo determinar que es una estructura liviana para dispositivos móviles de gran capacidad.



**Figura 43:** Consumo de memoria bandeja entrada del prototipo

Es fácil determinar que el mayor consumo de memoria se hace al momento de hacer el despliegue de los modelos 3D sobre el dispositivo móvil, se realizó el monitoreo de memoria sobre tres diferentes estudios para continuar observando el comportamiento de la memoria. En el primer estudio se tomó un hueso de tibia y peroné completamente sano (Leg.stl) tomando un total de 45.19MB de consumo de memoria, como se ilustra en la siguiente Figura 44:



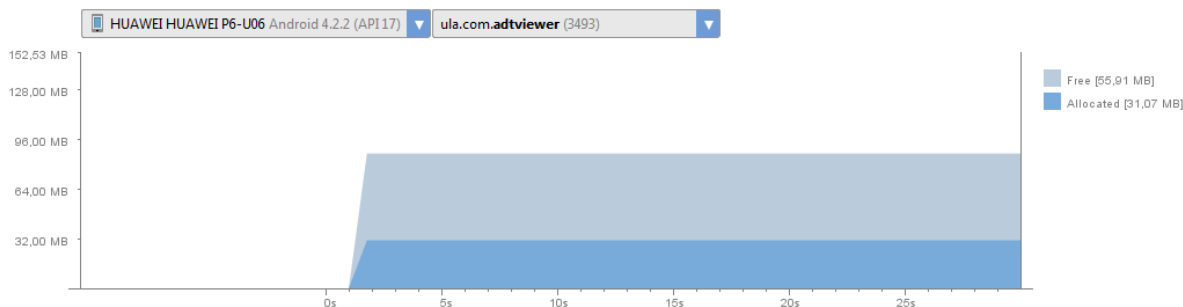
**Figura 44:** Consumo de memoria modelo *Leg.stl*

Un mayor detalle sobre el consumo de memoria para el anterior modelo, Android Studio permite hacer un desglose más detallado sobre cada una de las estructuras de datos implícitas dentro del prototipo para efectuar del despliegue tridimensional, tal como se muestra en la tabla 34:

	ID	Heap Size	Allocated	Free	% Used	# Objects	
	1	85,371 MB	39,445 MB	45,926 MB	46,20%	640.138	
Type	Count	Total Size	Smallest	Largest	Median	Average	
free	198.787	45,913 MB	16 B	8,629 MB	144 B	242 B	
data object	620.335	10,042 MB	16 B	1,031 KB	16 B	16 B	
class object	2.829	820,250 KB	168 B	41,039 KB	168 B	296 B	
1-byte array (byte[], boolean[])	672	25,001 MB	24 B	9,359 MB	672 B	38,096 KB	
2-byte array (short[], char[])	11.027	705,383 KB	24 B	28,023 KB	48 B	65 B	
4-byte array (object[], int[], float[])	5.255	2,906 MB	24 B	2,557 MB	40 B	579 B	
8-byte array (long[], double[])	20	6,727 KB	24 B	4,008 KB	136 B	344 B	
non-Java object	156	6,945 KB	16 B	480 B	32 B	45 B	

**Tabla 18:** Detalle consumo de memoria modelo *Spine\_Thoracic\_Lumbar\_half.stl*

Como segundo modelo se realizó el estudio para una columna lumbar (*Spine\_Thoracic\_Lumbar\_half.stl*) teniendo como resultado un aumento del consumo de la memoria a 31.07 MB debido a que es un estudio más detallado con mayor cantidad de información sobre cada una de sus partes como se encuentra en la Figura 45:



**Figura 45:** Consumo de memoria modelo *Spine\_Thoracic\_Lumbar\_half.stl*

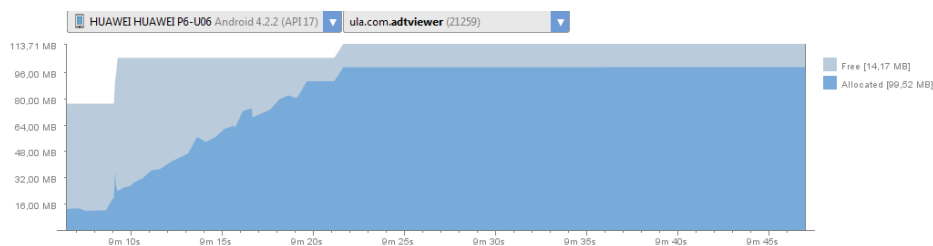
Continuando en el mismo esquema, para tener un mayor detalle sobre el consumo de memoria para el anterior modelo, se hace un desglose sobre cada una de las estructuras de datos implicadas dentro del prototipo en el sistema operativo para efectuar del despliegue tridimensional, tal como se muestra a en la tabla 35:

ID	Heap Size	Allocated	Free	% Used	# Objects		
1	62,719 MB	31,042 MB	31,677 MB	49,49%	462.889		
Type	Count	Total Size	Smallest	Largest	Median	Average	
free	139.756	31,665 MB	16 B	5,753 MB	144 B	237 B	
data object	443.102	7,339 MB	16 B	1,031 KB	16 B	17 B	
class object	2.822	818,977 KB	168 B	41,039 KB	168 B	297 B	
1-byte array (byte[], boolean[])	672	20,155 MB	24 B	6,542 MB	672 B	30,713 KB	
2-byte array (short[], char[])	11.021	705,016 KB	24 B	28,023 KB	48 B	65 B	
4-byte array (object[], int[], float[])	5.254	2,053 MB	24 B	1,704 MB	40 B	409 B	
8-byte array (long[], double[])	20	6,727 KB	24 B	4,008 KB	136 B	344 B	
non-Java object	155	6,797 KB	16 B	480 B	32 B	44 B	

**Tabla 19:** Detalle consumo de memoria modelo *Spine\_Thoracic\_Lumbar\_half.stl*

Por último en la misma categoría para el monitoreo de memoria se utilizó el modelo de una cabeza ósea (*Mandible.stl*) que es un estudio mucho más complejo donde se logra identificar en la siguiente Figura 46 un aumento considerable en el manejo de la memoria haciendo que la velocidad y movilidad sobre el objeto se vea afectada por la cantidad de datos.

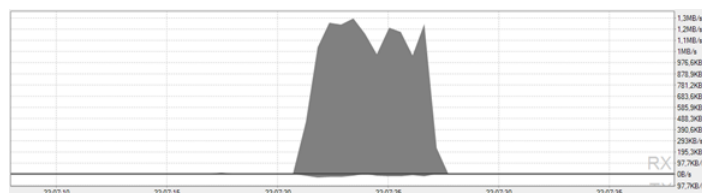




**Figura 46:** Consumo de memoria modelo Mandible.stl

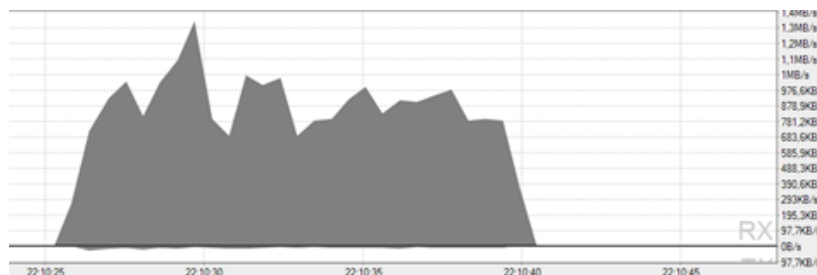
Otro aspecto importante para considerar es el consumo de la conexión de red para realizar las descargas de los modelos en el repositorio de archivos de la arquitectura propuesta. Al igual que el manejo de memoria, Android Studio permite monitorear el consumo de la red para las aplicaciones, en particular para el prototipo desarrollado, logrando establecer la velocidad de descarga de la aplicación en las condiciones de red del entorno de desarrollo dadas.

Para el modelo de prueba Spine\_Thoracic\_Lumbar\_half.stl que posee un tamaño de 6,700KB se logra observar en la siguiente Figura 47 que el tamaño máximo de descarga esta alrededor de los 1.3 MB/s, con un promedio de 1.1 MB/s en un tiempo de aproximadamente 8 segundos.



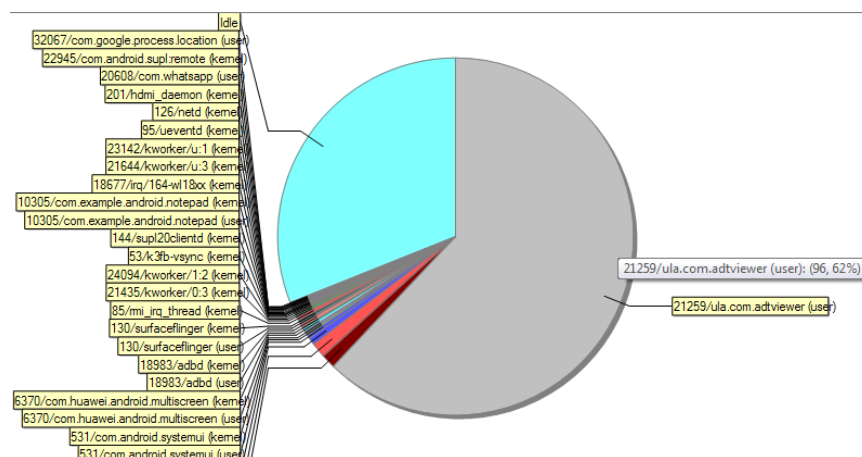
**Figura 47:** Velocidad de descarga modelo Spine\_Thoracic\_Lumbar\_half.stl

Para el modelo de la cabeza ósea que posee un tamaño de 12,221KB se logra observar en la Figura 48 que el tamaño máximo de descarga al igual que el modelo anterior esta alrededor de los 1.3 MB/s, con un promedio de 878 KB/s en un tiempo de aproximadamente 15 segundos, el cual va a depender del tráfico de la red en el momento de que el usuario efectuó la petición de visualizar el estudio en el dispositivo móvil.



**Figura 48:** Velocidad de descarga modelo Mandible.stl

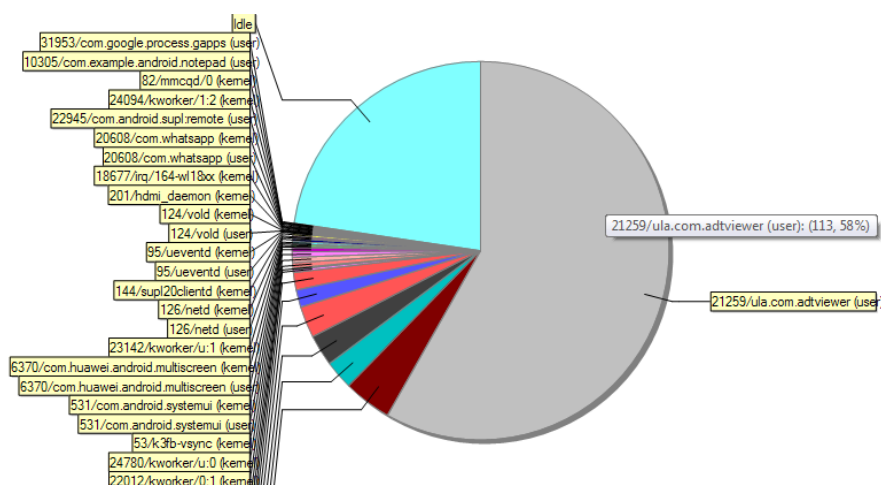
Otra variable importante a tomar en cuenta para el rendimiento del prototipo es el porcentaje de uso del CPU (Unidad Central de Procesamiento) con respecto a las demás aplicaciones instaladas en el dispositivo que cualquier usuario en particular pudiera estar usando al mismo tiempo más las propias del sistema operativo. Continuando con el modelo de columna lumbar y bajo el mismo entorno de desarrollo Android Studio 1.0 es viable poder visualizar el consumo del prototipo como se puede efectuar en la Figura 49, en donde el mismo (*ula.com.adtviewer*) consume un 62% al momento de hacer el despliegue tridimensional en el mismo dispositivo, a comparación de los demás procesos del usuario y kernel del sistema operativo.



**Figura 49:** Porcentaje de uso de CPU Spine\_Thoracic\_Lumbar\_half.stl

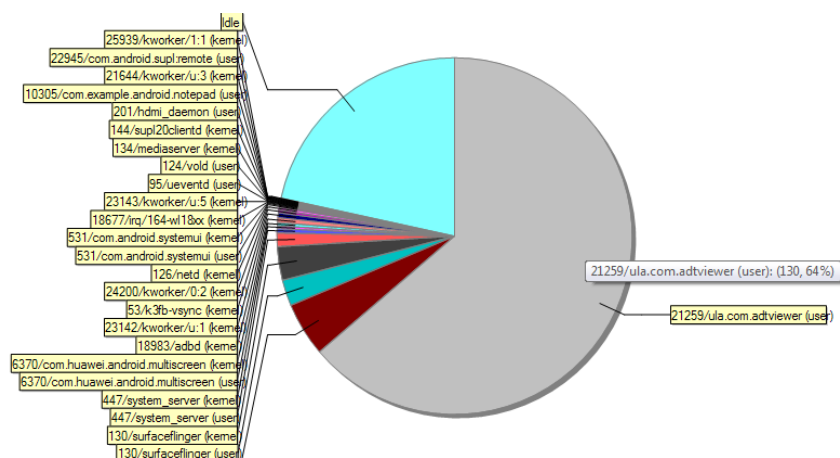
Para tener una visión más amplia sobre el comportamiento del prototipo bajo las condiciones dadas, se hace nuevamente la gráfica del porcentaje de uso de CPU pero esta vez para el modelo más liviano del hueso de la tibia y peroné, arrojando un 58% del total para el

proceso denominado *ula.com.adtviewer*, muy parecido al empleo del modelo anterior. La Figura 50 a continuación:



**Figura 50:** *Porcentaje de uso de CPU Leg.stl*

Por último es importante observar cómo se comporta el prototipo bajo modelos más pesados con mayor cantidad de detalle, este es el caso del modelo de la cabeza ósea, que como se encuentra en la Figura 51 aumenta a un 64% en la cantidad de uso de CPU con el proceso llamado *ula.com.adtviewer* bajo el identificador 21259 otorgado por el sistema operativo.



**Figura 51:** *Porcentaje de uso de CPU Mandible.stl*

Otra variable importante para medir el rendimiento del sistema es la frecuencia de reloj, el cual indica la velocidad con que el ordenador (en este caso el dispositivo móvil) efectúa las

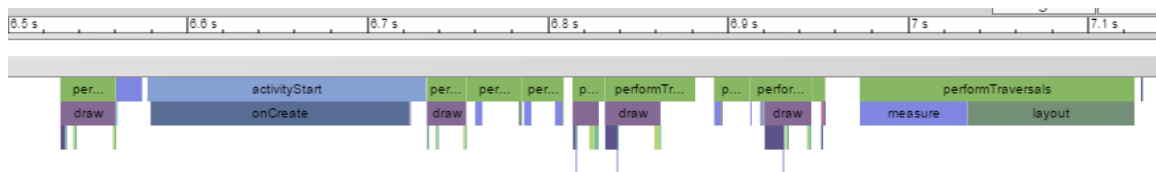
operaciones más básicas, es decir, la velocidad de funcionamiento del procesador medido en ciclos por segundos (medida en hercios), por lo que en general cuando se habla de frecuencia de reloj se hace referencia a la velocidad de CPU al momento de ejecutar cada una de sus aplicaciones en el sistema operativo (Stallings, 1997)

Android Studio cuenta con una herramienta denominada *Systrace* el cual ayuda a analizar como la aplicación dentro del entorno del sistema operativo Android se lleva a cabo, además permite recoger los datos de ejecución de código, mejorar el rendimiento y ver el sistema y las aplicaciones de la ejecución de procesos en una línea de tiempo común, permitiendo generar informes interactivos altamente detallados, en este caso para el modelo que se ha estado estudiando de hueso de la tibia y peroné, la Figura 52 de línea de tiempo desde los cero segundos hasta 29 segundos lleva a cabo una captura de la frecuencia de reloj al momento de comenzar a ejecutar el prototipo en el dispositivo móvil de desarrollo, ingresando las credenciales del usuario, seleccionando el modelo de estudio hasta generar el despliegue tridimensional. Allí se logra observar como a partir de los 7 segundos aproximadamente (tiempo en que se genera el listado de la Bandeja de Exámenes) el procesador trabaja con una frecuencia mayor, y a partir de los 20 segundos aproximadamente la etapa en la cual se levante dicho modelo en el equipo móvil, presentándose un aumento mayor en dicha frecuencia.



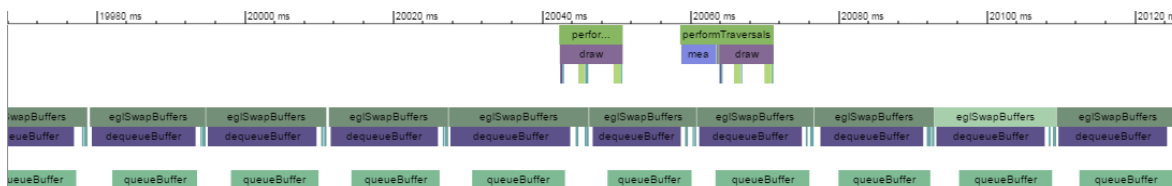
**Figura 52:** Frecuencia de reloj para el modelo *Leg.stl*

Como se mencionó anteriormente, a través del uso de la herramienta *Systrace* se logra obtener una traza de los métodos utilizados en el prototipo dentro de todo el entorno del sistema operativo, en este caso y bajo el anterior escenario, luego de introducir las credenciales del usuario y al momento de ingresarlas hasta la carga de la Bandeja de Exámenes, se despliega la Figura 53 con la línea de tiempo de ejecución de cada uno de los eventos llamados internamente en el código fuente del prototipo desarrollado.



**Figura 53:** Línea de tiempo métodos prototipo

Igualmente haciendo uso de la misma herramienta es posible visualizar la traza al momento de realizar la descarga de un modelo, en particular del estudio del hueso de la tibia y peroné, logrando determinar los tiempos de almacenamiento en búfer en milisegundos de los datos descargados del repositorio de archivos en formato STL. La Figura 54 a continuación ilustra lo antes expuesto:



**Figura 54:** Línea de tiempo descarga archivo Leg.stl

## CAPÍTULO VII

### CONCLUSIONES Y RECOMENDACIONES

#### 7.1 Conclusiones

La presente arquitectura propuesta consta de una plataforma computacional que permite el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos los cuales son capturados a través de estándares que incluyen los formatos de las imágenes, algoritmos de compresión, comunicaciones y estándares de red.

Gracias a las prestaciones que hoy en día poseen los dispositivos móviles es posible realizar sobre ellos un despliegue de gráficos 2D y 3D, por medio de estándares que definen una interfaz de programación de aplicaciones (API) multiplataforma para el procesamiento de gráficos para la posterior visualización del estudio a través de la arquitectura que contempla el tomógrafo, un sistema de base de datos, servidores web, repositorios de modelos de las imágenes DICOM y el prototipo básico desarrollado para representar de manera digital los estudios tridimensionales en formato Stereolithography (STL) sobre las características básicas mínimas de red y del dispositivo móvil para el mencionado despliegue.

El efecto de la arquitectura propuesta para el despliegue tridimensional en dispositivos móviles de datos generados por tomógrafos permite la reducción o eliminación de la película o discos compactos de los exámenes médicos para el posterior uso de los médicos especialistas en el área, además de la disponibilidad de los datos de forma inmediata posterior a la carga del modelo 3D en el sistema y para el su uso remoto desde en el dispositivo móvil, aumentando de la productividad y mayor satisfacción de los pacientes.

Los resultados muestran que es posible la implantación de la arquitectura y prototipo utilizando software libre como PHP, manejador de base de datos MYSQL en su versión gratuita

y servidor web de código abierto multiplataforma, usando el sistema operativo en su versión libre como Android que proporciona flexibilidad a terceros sobre cuestiones de desarrollo y concesión de licencias y soportado por la mayoría de dispositivos móviles creados hoy en día. Asimismo se cuenta con una API desarrollada en PHP lo cual representa una ventaja respecto a otras arquitecturas o sistemas creados con similares requerimientos ya que permite a futuros desarrolladores conectar sus aplicaciones no solamente a dispositivos móviles quedando abierta la posibilidad de utilizar otras tecnologías diferentes. Adicionalmente es totalmente escalable, permitiendo conectar mayor número de dispositivos, almacenar los modelos, registrar en una base de datos la información básica de la historia de los pacientes haciendo uso de un sistema web para hacer llegar al usuario del prototipo los datos generados por el tomógrafo y sin necesidad de navegadores web que sean compatibles con HTML5 quitando la carga de trabajo al servidor y reduciendo sus costos asociados.

Para estudios no tan complejos con mediana cantidad de datos en el formato Stereolithography y para dispositivos móviles de alta gama, los resultados indican que el prototipo presenta un alto consumo de memoria y uso de CPU, con una velocidad de descarga según la velocidad de la red a la memoria temporal del equipo y movilidad del modelo en la escena tridimensional creada permitiendo al usuario interactuar con el estudio 3D para la creación de diagnósticos precisos.

En los antecedentes de la investigación se presenta una serie de arquitecturas que pueden ser utilizadas para el despliegue tridimensional en dispositivos móviles, de las cuales, muestran ventajas y desventajas respecto a la arquitectura propuesta en el presente proyecto de investigación, el cual posee varias ventajas respecto a la plataforma presentada por Muhammad Fermi Pasha, Saravanesh Supramaniam y otros llamado “*An Android-based Mobile Medical Image Viewer and Collaborative*” ya que se encuentra desarrollada completamente en software libre, delimitando claramente los roles de cada usuario del sistema y siendo escalable según la cantidad de computo requerida, contando además, con un sistema web que facilita la gestión de la información sobre los exámenes de los pacientes para cada médico.

Otro antecedente presentado por Iuliana Ojog y Miguel Arias llamado “*m3DICOM: A Platform for mobile DICOM Visualization Based on X3D*” (traducido quiere decir m3DICOM, plataforma móvil para visualización de DICOM basada en X3D) posee como desventaja que el dispositivo móvil necesita de un navegador compatible con WebGL para la consulta de los modelos en HTML 5 y de una buena tarjeta de video para visualización 3D. Sin embargo, se necesita de software adicional del lado del servidor para convertir la estructura del directorio DICOM en el formato X3D y de un alto ancho de banda a través de HTTPS para la descarga de los archivos.

Oliver Goretzki en su investigación llamada “*3D Reconstruction of Medical Images Using Java3D*” (Reconstrucción 3D para imágenes médicas usando java3D) analiza las técnicas renderización comunes orientadas en el uso para el sector médico y presenta una propuesta para el renderizado utilizando java 3D, sin embargo, no posee una arquitectura que lleve a cabo la visualización 3D en dispositivos móviles.

Por último, Esmir Ramírez y Ernesto Coto presentan el proyecto denominado “Visualización web para datos médicos en 3D con software de código abierto”, usando los websockets para proveer una base en el desarrollo de aplicaciones para la web con despliegue tridimensional ejecutada sobre cualquier plataforma. A pesar de las ventajas que representan no cuenta con una base de datos de pacientes, médicos y exámenes que permita identificar los estudios por cada especialista. Además, cada movimiento del modelo genera un alto ancho de banda para la visualización en los dispositivos.

## **7.2 Recomendaciones.**

Posteriores investigadores pueden tomar en cuenta una mejora en el renderizado y uso de memoria para modelos tridimensionales con gran cantidad de cortes y detalles, ya que muchos estudios requieren de una alta precisión, lo que hace que al exportar las imágenes DICOM al



formato Stereolithography el archivo generado posea mayor cantidad de datos, generando una movilidad lenta en el despliegue de la escena 3D y dificultando el proceso de diagnóstico.

Adicionalmente es importante y de gran utilidad incluir los cortes de las imágenes DICOM sobre el modelo tridimensional desplegado en el prototipo, de modo que el usuario pueda observar la Figura 3D y el corte 2D simultáneamente (el corte que se imprime en la película), de forma que se contaría con mayor información útil para determinar cualquier patología.

Otro aspecto a mejorar para futuros investigadores en el proyecto propuesto es incluir la colaboración de diferentes usuarios o médicos sobre un mismo estudio para ayudar en la prestación de un mejor diagnóstico, planificación quirúrgica y seguimiento a los procedimientos a través de toma de decisiones en equipo.

## REFERENCIAS

- Franco, E. (2011). Análisis digital de imágenes topográficas sin contraste para la búsqueda de tumores cerebrales [Documento en línea]. Trabajo de grado de maestría, Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional. Disponible: <http://www.cs.cinvestav.mx/TesisGraduados/2011/TesisAdrianFranco.pdf> [Consulta: 2014, Agosto 28]
- Whitten, Jeffrey L. (1997). Análisis y diseño de sistemas de información. Santa Fé de Bogotá, COLOMBIA. McGraw - Hill/Irwin
- Centro Nacional de Tecnología de Información (2014). [Página web en línea]. Disponible: <http://cnti.gob.ve> [Consulta: 2014, septiembre 17]
- Comunidad Virtual de Anatomía Patológica (2014) [Documento en línea]. Disponible: [http://www.conganat.org/SEIS/is/is45/IS45\\_72.pdf](http://www.conganat.org/SEIS/is/is45/IS45_72.pdf) [Consulta: 2014, Septiembre 12]
- Muhammad Fermi Pasha, Saravanesh Supramaniam, Kwong Kuo Liang, Mohamad AmmarAmran, Bhavik Anil Chandra, and Mandava Rajeswari Universidad Sains Malasia (2012) [Documento en línea]. Disponible: <http://www.aicit.org/JDCTA/ppl/JDCTA900PPL.pdf> [Consulta: 2014, Septiembre 1]
- Ojog, I., Arias, M (2012) [Documento en línea]. Disponible: [http://www.thinkmind.org/download.php?articleid=etelemed\\_2012\\_2\\_40\\_40136](http://www.thinkmind.org/download.php?articleid=etelemed_2012_2_40_40136) [Consulta: 2014, Septiembre 3]
- Goretzki, O (2014) [Documento en línea]. Disponible en: [https://www.hochschule-trier.de/uploads/tx\\_rfttheses/GoretzkiOliver\\_Bachelor.pdf](https://www.hochschule-trier.de/uploads/tx_rfttheses/GoretzkiOliver_Bachelor.pdf) [Consulta: 2014, Septiembre 2]
- Ramírez, E., Coto, E (2012) [Documento en línea]. Disponible: <http://ccg.ciens.ucv.ve/~esmitt/publications/2012/jifi12.pdf> [Consulta: 2014, Septiembre 9]
- Wikipedia (2014). [Página web en línea]. Tomografía Disponible: <http://es.wikipedia.org/wiki/Tomograf%C3%ADa> [Consulta: 2014, septiembre 13]
- Mbits (2016). [Página web en línea]. Mray Disponible: <https://mbits.info/mray.php> [Consulta: 2016, Junio 20]
- JIUN Corporation's (2016). [Página web en línea]. SonicDICOM Disponible: <https://sonicdicom.com/features/> [Consulta: 2016, Junio 20]

- Rademacher, L (2014) ¿Qué tipos de tomógrafos existen? [Documento en línea].  
[http://www.ehowenespanol.com/tipos-tomografos-existen-lista\\_527721](http://www.ehowenespanol.com/tipos-tomografos-existen-lista_527721) [Consulta: 2014, Julio 19]
- Moyano e, Quiles f (1997) Técnica de compresión y codificación sobre imágenes médicas  
Escuela politécnica superior de Albacete. [Documento en línea].  
[http://www.uclm.es/ab/educacion/ensayos/pdf/revista13/13\\_26.pdf](http://www.uclm.es/ab/educacion/ensayos/pdf/revista13/13_26.pdf) [Consulta: 2015, Julio 10]
- Wikipedia (2014). [Página web en línea]. Dicom Disponible:  
<http://es.wikipedia.org/wiki/DICOM> [Consulta: 2014, Junio 19]
- Facultad de Informática de la Coruña (Sin Fecha) DICOM (Digital Imaging and Communications in Medicine) [Documento en línea].  
<http://www.fic.udc.es/files/asignaturas/58IB/DICOM-cas.pdf> [Consulta: 2014, Junio 8]
- Martínez, F (2006). Diseño y desarrollo de un PACS usando una librería de lectura y escritura del formato DICOM. Centro de Telemedicina de la Universidad Nacional de Colombia, Facultad de Medicina [Documento en línea].  
[http://www.unipamplona.edu.co/unipamplona/hermesoft/portalIG/home\\_1/recursos/tesis/contenidos/pdf\\_tesis/pdf\\_2/03052007/disenio\\_desarrollo\\_pacs.pdf](http://www.unipamplona.edu.co/unipamplona/hermesoft/portalIG/home_1/recursos/tesis/contenidos/pdf_tesis/pdf_2/03052007/disenio_desarrollo_pacs.pdf) [Consulta: 2014, Junio 23]
- Marshall Burns (1989) Automated Fabrication [Documento en línea].  
<http://www.ennex.com/~fabbers/StL.asp> [Consulta: 2015, Abril 8]
- De Giusti, Marisa Raquel; Lira, Ariel Jorge; Villarreal, Gonzalo Luján; Texier, José (2012). Las actividades y el planeamiento de la preservación en un repositorio institucional. [Documento en línea].  
[http://sedici.unlp.edu.ar/bitstream/handle/10915/26045/Documento\\_completo\\_\\_.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/26045/Documento_completo__.pdf?sequence=1) [Consulta: 2016, Junio 26]
- Basterretche, Juan Felix (2007). Dispositivos Móviles. [Documento en línea].  
<http://exa.unne.edu.ar/informatica/SO/tfbasterretche.pdf> [Consulta: 2016, Junio 26]
- Morillo, Julián David (2011). Introducción a los dispositivos móviles. [Documento en línea].  
[https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia\\_y\\_desarrollo\\_en\\_dispositivos\\_moviles/Tecnologia\\_y\\_desarrollo\\_en\\_dispositivos\\_moviles\\_%28Modulo\\_2%29.pdf](https://www.exabyteinformatica.com/uoc/Informatica/Tecnologia_y_desarrollo_en_dispositivos_moviles/Tecnologia_y_desarrollo_en_dispositivos_moviles_%28Modulo_2%29.pdf) [Consulta: 2016, Junio 26]

- Padrón, J. (1998). La Estructura de los Procesos de Investigación. Disponible en:  
<http://dip.una.edu.ve/mae/978investigacioneducativa/paginas/Lecturas/UNIDAD%204/Padron-LaEstructuradelosProcesosdeInvestigacion.pdf>. [Consulta: 2014, Septiembre 17]
- Sabino, C. (1992). El proceso de investigación. [Libro en línea]. Caracas, Venezuela. Editorial Panapo. Disponible:  
<http://www.iutep.tec.ve/uftp/images/Descargas/materialwr/libros/CarlosSabino-ElProcesoDeInvestigacion.PDF>. [Consulta: 2014, Septiembre 17]
- Boehm, B. (1998). [Documento en línea]. Disponible:  
<http://csse.usc.edu/csse/TECHRPTS/1988/usccse88-500/usccse88-500.pdf> [Consulta: 2014, Octubre 20]
- Project Management Institute, “A guide to the Project Management Body of Knowledge – PMBOK”, Third Edition, 2004
- Santos, W. (2014) [Documento en línea]. Disponible: <http://m0del0espiral.blogspot.com/>  
 [Consulta: 2014, Octubre 20]
- Ferré, X y Sánchez, M (2011) [Libro en línea]. Desarrollo Orientado a Objetos con UML  
<http://www.uv.mx/personal/maymendez/files/2011/05/umlTotal.pdf> [Consulta: 2015, Abril 2]
- Shoji Kaoru (2011) [Página web en línea]. STL Viewer for Android. Disponible:  
<http://www.thingiverse.com/thing:9656> [Consulta: 2016, Julio 18]
- Android Developers (2015) [Página web en línea]. Analyzing Display and Performance.  
 Disponible: <http://developer.android.com/tools/debugging/systrace.html> [Consulta: 2015, Marzo 11]
- Huawei Technologies Co (2014) [Página web en línea]. Ascend P6. Disponible:  
<http://www.huaweispain.com/smartphones/huawei-ascend-p6/> [Consulta: 2015, Marzo 1]
- MakerBot Industries, LLC (2015) [Página web en línea]. Disponible:  
<https://www.thingiverse.com/about> [Consulta: 2015, Marzo 10]
- Stallings William (1997) Sistemas Operativos, Segunda Edición. Madrid, España. Prentice Hall.

