**UNIVERSIDAD AUTÓNOMA GABRIEL RENÉ MORENO**

**FACULTAD DE INGENIERÍA EN CIENCIAS   
DE LA COMPUTACIÓN Y** **TELECOMUNICACIONES**

**SOFTWARE PARA LA RECEPCIÓN DE PACIENTES ASEGURADOS EN EL ÁREA DE EMERGENCIAS CON RECONOCIMIENTO FACIAL USANDO REDES NEURONALES PARA LA CLÍNICA MELENDREZ**

**INTEGRANTES:**

* Mariana Hurtado Soliz
* Moises Leonardo Mogiano Gutierrez

**CARRERA:**

* Ingeniería en Sistemas

**MATERIA:**

* Taller de Grado

**NOMBRE DEL DOCENTE:**

Javier Alanoca Gutiérrez, PhD

Julio 2023

Santa Cruz de la Sierra – Bolivia

**Tabla de Contenido**

[**CAPÍTULO I** 5](#_Toc139561314)

[**1.1.** **Planteamiento del Problema** 5](#_Toc139561315)

[**1.1.2. Situación Deseada** 5](#_Toc139561316)

[**1.1.3. Objeto de Investigación** 6](#_Toc139561317)

[**1.2.** **Objetivo** 6](#_Toc139561318)

[**1.2.1** **Objetivo General** 6](#_Toc139561319)

[**1.2.2** **Objetivo Específico** 6](#_Toc139561320)

[**1.3.** **Metodología** 7](#_Toc139561321)

[**1.3.1. Fase de Inicio** 7](#_Toc139561322)

[**1.3.2. Fase de Elaboración** 8](#_Toc139561323)

[**1.3.3. Fase de Construcción** 8](#_Toc139561324)

[**1.4. Alcance** 8](#_Toc139561325)

[**1.4.1. Módulo de Paciente** 9](#_Toc139561326)

[**1.4.2. Módulo de Personal Médico** 9](#_Toc139561327)

[**1.4.3. Módulo de Reconocimiento de Pacientes** 9](#_Toc139561328)

[**1.4.4. Módulo de Emergencias** 9](#_Toc139561329)

[**1.4.5. Módulo de Historia Clínica** 9](#_Toc139561330)

[**CAPÍTULO II** 10](#_Toc139561331)

[**2.1. Detección Facial** 10](#_Toc139561332)

[**2.2. Proceso de Detección Facial** 11](#_Toc139561333)

[**2.3. Redes Neuronales Convolucionales** 12](#_Toc139561334)

[**CAPÍTULO III** 14](#_Toc139561335)

[**3.1. Antecedentes** 14](#_Toc139561336)

[**3.2. Misión** 14](#_Toc139561337)

[**3.3. Visión** 14](#_Toc139561338)

[**3.4. Estructura Organizacional** 15](#_Toc139561339)

[**3.5. Infraestructura** 15](#_Toc139561340)

[**3.6. Procedimiento de Identificación de Paciente** 16](#_Toc139561341)

[**CAPÍTULO IV** 17](#_Toc139561342)

[**4.1. Introducción** 17](#_Toc139561343)

[**4.1.1. Propósito** 17](#_Toc139561344)

[**4.1.2. Ámbito de Sistema** 17](#_Toc139561345)

[**4.1.3. Definiciones, Acrónimos y Abreviaturas** 18](#_Toc139561346)

[**4.1.4. Referencia** 18](#_Toc139561347)

[**4.2. Descripción General** 18](#_Toc139561348)

[**4.2.1. Perspectiva del Proyecto** 19](#_Toc139561349)

[**4.2.2. Funcionalidades del Sistema** 19](#_Toc139561350)

[**4.2.3. Características de los Usuarios** 19](#_Toc139561351)

[**4.2.4. Restricciones** 20](#_Toc139561352)

[**4.3. Requisitos Específicos** 20](#_Toc139561353)

[**4.3.1. Requisitos Funcionales** 20](#_Toc139561354)

[**4.3.2. Requisitos No Funcionales** 21](#_Toc139561355)

[**CAPÍTULO V** 23](#_Toc139561356)

[**5.1. Identificación de Actores** 23](#_Toc139561357)

[**5.2. Identificación de Casos de Uso** 23](#_Toc139561358)

[**5.3. Priorización de Casos de Uso** 24](#_Toc139561359)

[**5.4. Detalle de Casos de Uso** 25](#_Toc139561360)

[**5.4.1. CU01 Gestionar Personal Médico** 25](#_Toc139561361)

[**5.4.2. CU02 Gestionar Pacientes** 26](#_Toc139561362)

[**5.4.3. CU03 Gestionar Emergencias** 27](#_Toc139561363)

[**5.4.4. CU04 Identificación de Pacientes** 28](#_Toc139561364)

[**5.4.5. CU05 Designación de Análisis** 29](#_Toc139561365)

[**5.4.6. CU06 Gestionar Historia Médica** 30](#_Toc139561366)

[**5.5. Diagrama General de Casos de Uso** 31](#_Toc139561367)

[**CAPÍTULO VI** 32](#_Toc139561368)

[**6.1. Análisis de Paquetes** 32](#_Toc139561369)

[**6.1.1. Identificación de Paquetes** 32](#_Toc139561370)

[**6.2. Análisis de Arquitectura** 33](#_Toc139561371)

[**6.2.1. Vista de Paquetes Gestión de Usuario** 33](#_Toc139561372)

[**6.3. Análisis de Casos de Uso** 35](#_Toc139561373)

[**6.3.1. Diagrama de Comunicación** 35](#_Toc139561374)

[**CAPÍTULO VII** 38](#_Toc139561375)

[**7.1. Diseño de la Arquitectura** 38](#_Toc139561376)

[**7.2. Diagrama de Secuencia** 39](#_Toc139561377)

[**7.3. Diseño de Datos** 45](#_Toc139561378)

[**7.3.1. Diseño Conceptual** 45](#_Toc139561379)

[**7.3.2. Diseño Lógico** 46](#_Toc139561380)

[**7.3.3. Diseño Físico** 47](#_Toc139561381)

[**7.4. Diseño de Interfaz** 50](#_Toc139561382)

[**CAPÍTULO VIII** 55](#_Toc139561383)

[**8.1. Implementación de la Arquitectura** 55](#_Toc139561384)

[**8.1.1. Implementación de Componentes** 55](#_Toc139561385)

[**8.2. Herramientas de Desarrollo** 56](#_Toc139561386)

[**8.2.1. Sistema Operativo** 56](#_Toc139561387)

[**8.2.2. Lenguaje de Programación** 57](#_Toc139561388)

[**8.2.3. Base de Datos** 57](#_Toc139561389)

[**8.2.4. Herramientas Case** 58](#_Toc139561390)

[**CAPÍTULO IX** 59](#_Toc139561391)

[**9.1. Costos** 59](#_Toc139561392)

[**9.2. Riesgos** 60](#_Toc139561393)

[**9.2.1. Identificación de Riesgos** 60](#_Toc139561394)

[**9.2.2. Plan de Aversión** 61](#_Toc139561395)

[**9.3. Seguridad** 63](#_Toc139561396)

[**9.3.1. Seguridad Física** 63](#_Toc139561397)

[**9.3.2. Seguridad Lógica** 63](#_Toc139561398)

[**CAPÍTULO X** 65](#_Toc139561399)

[**10.1. Planificación de Pruebas** 65](#_Toc139561400)

[**10.2. Diseño de Pruebas** 66](#_Toc139561401)

[**10.2.1. Identificación y Estructuración de Procedimiento de Prueba** 66](#_Toc139561402)

[**10.3. Pruebas Realizadas** 67](#_Toc139561403)

[**CONCLUSIÓN** 77](#_Toc139561404)

[**BIBLIOGRAFÍA** 78](#_Toc139561405)

**CAPÍTULO I**

INTRODUCCIÓN

La clínica Melendres ha experimentado dificultades en su funcionamiento debido al incremento en la cantidad de pacientes. Esto ha generado problemas cuando el personal médico identifica a los pacientes en situaciones de emergencia. Lo cual es preocupante ya que repercute en el acceso a la información médica del paciente como en la rapidez para actuar en este tipo de situaciones.

Con el objetivo de buscar una solución, se plantea diseñar una aplicación móvil que permita la identificación sencilla y precisa de los pacientes en el área de emergencias mediante reconocimiento facial. Esto agilizará el proceso de obtención de información y brindará una atención más eficiente y crucial.

## **1.1.** **Planteamiento del Problema**

**1.1.1.** **Situación Problemática**

La clínica Melendres se ha enfrentado a dificultades en su funcionamiento debido al incremento en la cantidad de pacientes asegurados. Esto ha ocasionado problemas al momento que el personal médico identifica a los pacientes en el área de emergencia. De tal manera alargando los tiempos en la obtención de información médica relevante de los pacientes, lo que es preocupante dada la importancia que tiene la información a la hora de brindar diagnósticos precisos como también el tiempo para actuar en este tipo de situaciones.

### **1.1.2. Situación Deseada**

Cuando los pacientes lleguen en estado de emergencia, el personal médico va a poder identificarlos de manera sencilla. Asimismo, obteniendo información médica relevante del paciente, como también ahorrando tiempo el cual es crucial en situaciones de emergencia, consecuentemente los doctores podrán actuar con mayor cautela en base a la información relevante de la historia médica.

### **1.1.3. Objeto de Investigación**

Análisis del proceso de reconocimiento para optimizar la identificación de los pacientes en el área de emergencias.

** **Pacientes** **Clínica** **Identificación** **Búsqueda Historia**

*Ilustración 1: Objeto de Investigación*

*Fuente (Elaboración Propia)*

## **1.2.** **Objetivo**

### **1.2.1** **Objetivo General**

Diseñar una aplicación móvil para mejorar la eficiencia en la identificación y recepción de pacientes en el área de emergencias de la clínica Meléndrez mediante la implementación de reconocimiento facial a través de inteligencia artificial con redes neuronales convolucionales.

### **1.2.2** **Objetivo Específico**

* Estudiar el problema actual que tiene la clínica Melendres, en el área de emergencia, encontrar sus causantes y encontrar la manera de solucionarlo.
* Analizar las características del proceso de identificación de pacientes en estado de emergencia en una clínica.
* Definir los requisitos funcionales y no funcionales del software a través de los requerimientos.
* Elaborar una arquitectura del software para que este sea escalable y con bajo acoplamiento.
* Realizar el diseño del sistema con el objetivo de planificar adecuadamente la implementación y asegurar la satisfacción del usuario.
* Analizar el software con el propósito de validar el cumplimiento de la alineación de requisitos y expectativas.
* Implementar el software diseñado.
* Probar el software con el objetivo de encontrar y corregir fallas en la misma.

## **1.3.** **Metodología**

La metodología que hemos adoptado para el proyecto es el Proceso Unificado de Desarrollo de Software (PUDS) de Jacobson y Booch (2001), que consta de cuatro fases: inicio, elaboración, construcción y transición. Utilizaremos el Lenguaje Unificado de Modelado (UML) como lenguaje de representación visual para mostrar las fases, flujos de trabajo y actividades del PUDS.

### **1.3.1. Fase de Inicio**

**Captura de requisitos**

❖ Requisitos funcionales y no funcionales

❖ Lista de actores y casos de uso

❖ Detalle de casos de uso

❖ Prototipo de la interfaz de los casos de uso

**Resultado**

❖ Modelo de Casos de uso

**1.3.2. Fase de Elaboración**

**Actividades**

❖ Diseño de arquitectura

❖ Diseño de casos de uso

**Resultado**

❖ Descripción del diseño de software

### **1.3.3. Fase de Construcción**

**Implementación**

Utilizaremos el UML (Lenguaje Unificado de Modelado) como lenguaje de representación visual.

**Fase de Transición**

En esta fase final, el programa debe estar listo para ser probado, instalado y utilizado por el cliente sin ningún problema.

**1.4. Alcance**

Los módulos que se desarrollaran en el siguiente proyecto son los siguientes:

* Módulo de Paciente
* Módulo de Personal Médico
* Módulo de Reconocimiento de Identificación de Pacientes
* Módulo de Emergencia
* Módulo de Historia Clínica

### **1.4.1. Módulo de Paciente**

En este módulo se gestionará la información del paciente del área de emergencias.

### **1.4.2. Módulo de Personal Médico**

En este módulo se gestionará la información del personal médico del área de emergencias, como los roles y privilegios de estos.

### **1.4.3. Módulo de Reconocimiento de Pacientes**

En este módulo de reconocimiento se tomarán fotografías del rostro de los pacientes del área de emergencia para reconocer y obtener sus datos de manera sencilla en caso de ser asegurado recepcionará su llegada, caso contrario no este asegurado se gestionará como nuevo usuario.

### **1.4.4. Módulo de Emergencias**

Este módulo se gestionará las emergencias que hayan sido recepcionadas, el personal médico y administrativo podrá ver la información detallada de la misma, así mismo como la del paciente atendido.

### **1.4.5. Módulo de Historia Clínica**

En este módulo se podrá administrar la historia médica de los pacientes asegurados y nuevos, para posteriormente ser usada al momento que personal médico requiera dar algún tratamiento u operación.

**CAPÍTULO II**

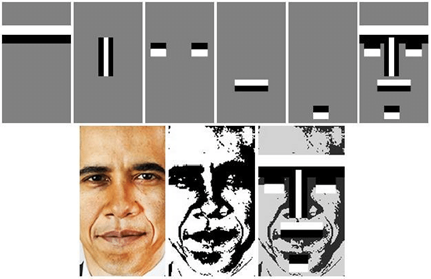
PROCESOS DE DETECCIÓN FACIAL Y REDES NEURONALES CONVOLUCIONALES

**2.1. Detección Facial**

La detección facial es el proceso mediante el cual el sistema localiza la posición de los rostros de personas en una imagen o fotograma (Turk & Pentland, 1991). Aunque en los inicios del reconocimiento facial, el proceso de localización de la cara en la imagen tenía que ser llevada a cabo manualmente, en la actualidad existen diferentes técnicas de detección automática, con diferentes fundamentos (Zhao et al., 2003). Aunque en algunas ocasiones es difícil clasificar algunos métodos, existen autores que han realizado clasificaciones para diferenciar entre algunos métodos y otros, según su fundamento (Yang, Kriegman, & Ahuja, 2002). En su trabajo, se distinguen los métodos que se basan en el conocimiento, los basados en características invariantes, los que hacen uso de plantillas o modelos y los que se basan en apariencia.

Como no es posible nombrar y describir todos los métodos de detección existentes debido a la gran variedad de algoritmos (que además a su vez presentan variantes), se van a nombrar tres que se han considerado de relevancia en este proyecto. Para los dos primeros se hará un breve repaso de su funcionamiento y para el último se entrará en profundidad ya que será el método que se utilice para implementar el sistema de reconocimiento facial de este proyecto.

## **2.2. Proceso de Detección Facial**

El algoritmo propuesto por Viola, P., & Jones, M. (2001) fue la primera técnica de detección facial que presentaba una rapidez y precisión apta para su uso en tiempo real. Este método se basa en utilizar una serie de características denominadas Haar-like features, que son obtenidas a partir del producto escalar entre una imagen y un patrón simple del mismo tamaño que la imagen. Dado un conjunto de Haar-like features y un conjunto de imágenes positivas y negativas, Viola y Jones implementaron una variante de AdaBoost para el entrenamiento de su clasificador.

*Ilustración 3:* *Haar-like features*

*Fuente: Kadir y Kamaruddin (2013). “A comparative study between LBP and Haar-like features for Face Detection ".*

AdaBoost se trata de un algoritmo de aprendizaje creado por Freund y Schapire (1997) que consiste en obtener un clasificador fuerte a partir de varios clasificadores débiles que fijarán unos umbrales que, puestos en forma de cascada, formarán la región delimitada para una clase. Una vez haya sido entrenado este clasificador, será posible distinguir si el resultado de calcular las Haar-like features se corresponde con un rostro en cierto lugar de la imagen (Viola & Jones, 2001).

Detección

Procesado

Extracción de características

Base de datos

Comparación y clasificación

*Ilustración 2: Diagrama de un Sistema de Reconocimiento Facial*

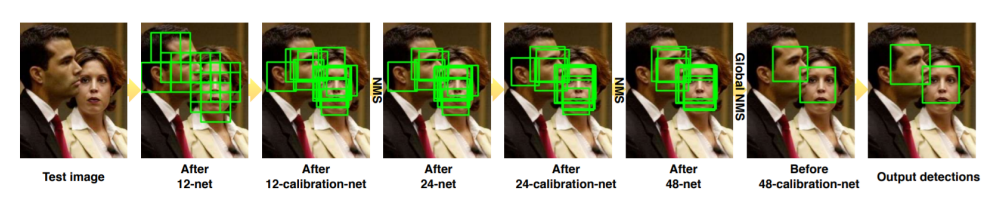
*Fuente (Elaboración Propia)*

## **2.3. Redes Neuronales Convolucionales**

Las redes neuronales convolucionales (CNN, por sus siglas en inglés) son una herramienta con múltiples aplicaciones, y aunque no se van a utilizar en este proyecto para la detección facial, se hace un breve repaso informativo de un método que utiliza estas redes para detectar caras en imágenes.

En comparación con la clasificación de rostros en el último paso del reconocimiento facial, la detección de rostros en imágenes utilizando redes neuronales es una tarea más difícil de llevar a cabo (Rowley, Baluja y Kanade, 1998). En la actualidad, existen diferentes trabajos en los que se han implementado redes neuronales para la detección facial (Rowley, Baluja y Kanade, 1998; Lin, Kung y Lin, 2004), y en este apartado se hará un breve repaso del trabajo presentado por Li, Lin, Shen, Brandt y Hua (2015), en el que se propone una serie de CNN en forma de cascada para detectar rostros directamente de la propia imagen, aprendiendo automáticamente las características en las que se tiene que basar.

En su trabajo, Li et al. (2015) describen seis CNN, de las cuales tres son de detección y tres de calibración, estando intercaladas unas con otras. Las redes de detección, especializadas en 12, 24 y 48 píxeles de resolución respectivamente son utilizadas para localizar posibles caras en la imagen. Luego, la correspondiente red de calibración se encargará de procesar las ventanas detectadas en las imágenes para ajustar su tamaño y localización para acercarse a una cara potencial cercana. En definitiva, se encarga de calibrar los recuadros que rodean las caras para ajustarlos a la cara que es posible que puedan contener.

De esta forma, la cascada irá acercándose a las localizaciones de los rostros en la imagen hasta detectarlos completamente.

*Ilustración 4: Las Redes Convolucionales en Cascada para Entrenamiento*

*Fuente: Li, H., Lin, Z., Shen, X. y Brandt, J. (2015). "A convolutional neural network cascade for face detection".*

**CAPÍTULO III**

CLÍNICA MELENDRES

## **3.1. Antecedentes**

La Clínica Médica Melendres es una institución fundada en 1994 por la familia Melendres en los Valles Cruceños de Bolivia. Desde entonces, ha experimentado un continuo crecimiento hasta llegar a situarse finalmente en la ciudad oriental de Santa Cruz, donde actualmente cuenta con más de 22 especialidades médicas. Gracias a su calidad de atención y variedad de servicios, se ha posicionado como una gran clínica de calidad en la ciudad.

## **3.2. Misión**

Somos una clínica de especialidades médicas con principios y valores, con la vocación de salvar vidas, brindando una salud accesible a la población, con calidad y calidez de atención y respuesta inmediata.

## **3.3. Visión**

Ser Líderes en salud con accesibilidad y calidad para todos, brindando confianza y bienestar a nuestros clientes con alta tecnología y estándares internacionales.

## **3.4. Estructura Organizacional**

*Ilustración 5: Estructura Organizacional*

*Fuente (Elaboración Propia)*

## **3.5. Infraestructura**

*Ilustración 6: Ubicación Geográfica*

*Fuente (Google Maps)*

Está ubicada en el 3er anillo interno de la Av. Grigotá, Doble vía a la Guardia Nro. 2450

## **3.6. Procedimiento de Identificación de Paciente**

*Ilustración 7: Proceso de Identificación de Paciente*

*Fuente (Elaboración Propia)*

# **CAPÍTULO IV**

**ESPECIFICACIÓN DE REQUISITOS SEGÚN EL ESTÁNDAR IEEE 830**

## **4.1. Introducción**

En la presente sección del documento se procederá a detallar los requisitos del software a través de distintos mecanismos, como ser entrevistas, encuestas u otros. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos de Software ANSI/IEEE 830,1998.

### **4.1.1. Propósito**

El presente documento tiene como objetivo definir las funcionalidades y restricciones necesarias para el desarrollo de un software de Recepción de Pacientes en el Área de Emergencias con Reconocimiento Facial, utilizando Redes Neuronales Convolucionales, para la Clínica Meléndrez. Este documento servirá como canal de comunicación entre el personal médico y la administración del hospital para una mejor comprensión del software. El documento está dirigido al personal médico y a la administración.

### **4.1.2. Ámbito de Sistema**

El ámbito del sistema se centra en la recepción de pacientes en el área de emergencias, donde el software utiliza reconocimiento facial a través de redes neuronales convolucionales para optimizar el proceso de atención médica. El objetivo es brindar una atención rápida y eficiente al paciente, lo cual es fundamental en casos de emergencia.

### **4.1.3. Definiciones, Acrónimos y Abreviaturas**

|  |  |
| --- | --- |
| ***Nombre*** | ***Descripción*** |
| **Personal médico** | Pueden ser Enfermeras o Doctores |
| **Administrador** | Persona que podrá registrar los datos del paciente |
| **CNN** | Las redes neuronales convolucionales |
| **ERS** | Especificación de requisitos software |
| **IA** | Inteligencia Artificial |
| REF | Reconocimiento Facial |

*Ilustración 8: Definiciones, Acrónimos y Abreviaturas*

*Fuente (Elaboración Propia)*

### **4.1.4. Referencia**

Méndez, G (22 de octubre 2008) Especificación de Requisitos según el Estándar de la IEEE 830.

## **4.2. Descripción General**

En la presente sección del documento se procederá a detallar los requisitos del software a través de distintos mecanismos, como ser entrevistas, encuestas u otros. Esta especificación se ha estructurado basándose en las directrices dadas por el estándar IEEE Práctica Recomendada para Especificaciones de Requisitos de Software ANSI/IEEE 830,1998.

### **4.2.1. Perspectiva del Proyecto**

El software para la recepción de pacientes en el área de emergencias con reconocimiento facial estará bajo el framework de Flutter 3 para la realización de sus funciones principales como sistema de gestión de clínica y el reconocimiento de usuarios.

### **4.2.2. Funcionalidades del Sistema**

El sistema proporcionara el soporte de las siguientes funcionalidades:

* **F1. Registro de Paciente:** El personal administrativo podrá registrar al paciente por medio de la aplicación.
* **F2. Registro de Personal Médico:** El personal administrativo podrá registrar al personal médico por medio de la aplicación.
* **F3. Reconocimiento Facial para Pacientes:** El personal médico podrá subir una imagen desde su teléfono a la aplicación e identificar al paciente.
* **F4. Gestionar Emergencias:** El personal administrativo u médico podrá gestionar la lista de emergencias recepcionadas.
* **F5. Gestionar Historia Médica:** El personal médico podrá acceder a la historia médica del paciente.
* **F6. Designación de Análisis:** El personal médico adjunta los análisis requeridos a realizarse.

### **4.2.3. Características de los Usuarios**

|  |  |  |
| --- | --- | --- |
| **Usuario** | **Características** | **Descripción** |
| **Personal Administrativo** | Conocimientos de administración de personal.   Conocimientos básicos de computación. | Es el encargado de administrar el sistema deberá tener mínimo conocimientos informáticos. |
| **Personal Medico** | No necesitan conocimientos técnicos | Tendrá acceso a la historia médica del paciente |

*Ilustración 9: Características de los Usuarios*

*Fuente (Elaboración Propia)*

### **4.2.4. Restricciones**

Las restricciones son las siguientes:

**4.2.4.1 Políticas Reguladoras**

La aplicación se desarrollará mediante el framework de Flutter disponible en GitHub bajo la licencia MIT.

**4.2.4.2 Requisitos del Lenguaje**

El sistema debe controlar los permisos que tiene cada usuario para su accesibilidad de una manera correcta, de tal forma que pueda acceder.

**4.2.4.2 Suposiciones y Dependencias**

Se han definido las siguientes suposiciones:

* Estará implementada en un host virtual llamado Digital Ocean que brindará soporte a la API que consumirá la App Móvil.
* Los requerimientos del sistema no cambiaran durante el desarrollo del software.
* Para la comunicación entre el sistema deberá tener acceso a internet.

## **4.3. Requisitos Específicos**

### **4.3.1. Requisitos Funcionales**

**Personal Administrativo**

**RF1. Registra Datos de Pacientes:** El personal administrativo tendrá acceso para registrar la información de los datos necesarios del paciente.

**RF2. Registra Datos de Personal Médico:** El personal administrativo tendrá acceso para registrar la información de los datos necesarios del personal médico.

**RF3: Completar Información de Emergencias:** El personal médico podrá completar la información de las emergencias recepcionadas.

**Personal Médico**

**RF4: Visualizar Emergencias:** El personal médico podrá gestionar las emergencias recepcionadas.

**RF5: Realiza Consulta con Reconocimiento Facial:** El personal médico podrá acceder a la historia médica del paciente y sus datos.

**RF6: Designación de Análisis:** El personal médico podrá designar los análisis requeridos para el paciente.

**RF7: Gestionar Historia Medica:** El personal médico podrá designar actualizar la información de la historia médica de los pacientes.

### **4.3.2. Requisitos No Funcionales**

A continuación, se citarán los requisitos no funcionales:

1. **Requisitos del rendimiento**

El tiempo de respuesta de la App Móvil no deberá ser mayor a 8 segundos.

1. **Seguridad**

Garantizar las autentificaciones como filtros de seguridad, ya sea en inicio de sesión o manejo de información del paciente.

1. **Fiabilidad**

El software deberá tener una interfaz sencilla para el manejo del usuario.

1. **Disponibilidad**

La disponibilidad del software estará disponible las 24 horas del día, brindando toda la información necesaria en el momento que el usuario lo requiera.

1. **Mantenibilidad**

* El software deberá estar basado en un estándar de programación.
* Deberá disponer de un manual de usuario sobre las operaciones del sistema, para que cualquier personal encargado pueda realizar mantenimiento del software.

1. **Portabilidad**

* El software debe ser compatible con diferentes dispositivos con versiones variables de Android.
* Se debe garantizar que la interfaz de usuario sea adaptable a pantallas de diferentes tamaños y resoluciones.

1. **Escalabilidad**

* El software debe ser escalable para poder manejar un crecimiento futuro en términos de usuarios, datos y funcionalidades.
* Debe estar preparado para soportar un aumento de carga sin comprometer su rendimiento.

# **CAPÍTULO V**

**CAPTURA DE REQUISITOS**

## **5.1. Identificación de Actores**

*Ilustración 10: Identificación de Actores*

*Fuente (Elaboración Propia)*

## **5.2. Identificación de Casos de Uso**

|  |  |
| --- | --- |
| Identificador | Nombre |
| CU01 | Gestionar Personal Médico |
| CU02 | Gestionar Pacientes |
| CU03 | Gestionar Emergencias |
| CU04 | Identificación de Pacientes |
| CU05 | Designación de Análisis |
| CU06 | Gestionar Historia Médica |

*Ilustración 11: Identificación de casos de usos*

*Fuente (Elaboración Propia)*

## **5.3. Priorización de Casos de Uso**

|  |  |  |
| --- | --- | --- |
| Identificador | Nombre | Prioridad |
| CU01 | Gestionar Personal Médico | Media |
| CU02 | Gestionar Pacientes | Media |
| CU03 | Gestionar Emergencias | Media |
| CU04 | Identificación de Pacientes | Alta |
| CU05 | Designación de Análisis | Baja |
| CU06 | Gestionar Historia Médica | Alta |

*Ilustración 12: Priorización de Casos de Usos*

*Fuente (Elaboración Propia)*

## **5.4. Detalle de Casos de Uso**

### **5.4.1. CU01 Gestionar Personal Médico**

*  
Ilustración 13: Gestionar Personal Médico*

*Fuente (Elaboración Propia)*

|  |  |
| --- | --- |
| **Casos de Uso** | **CU01: Gestionar Personal Médico** |
| **Propósito** | Permite gestionar al personal médico. |
| **Descripción** | Permite gestionar al personal médico, doctores y enfermeras, registrando, modificando y eliminando sus datos dentro del software |
| **Actores** | Personal Administrativo, Administrador |
| **Actor Inicial** | Personal Administrativo |
| **Pre-Condición** | Ninguna |
| **Proceso** | **1.- Insertar Personal médico.**  1.1 Agregar la información.  1.2. Definir si es doctor(a) o Enfermero(a)  1.2. Insertar los datos  1.3. Guardar datos  **2.- Modificar Personal medico**  2.1. Seleccionar Personal medico  2.2. Modificar datos  2.3. Guardar cambios  **3.- Eliminar Personal medico**  3.1. Seleccionar Personal medico  3.2. Confirmar eliminación  3.3. Guardar cambios |
| **Postcondición** | Ninguna |
| **Excepción** | 1. Datos incorrectos.  2. No se seleccionó ningún personal médico.  3. No se confirmó la eliminación. |

### **5.4.2. CU02 Gestionar Pacientes**



*Ilustración 14: Gestionar Pacientes*

*Fuente (Elaboración Propia)*

|  |  |
| --- | --- |
| **Casos de Uso** | **CU02: Gestionar Pacientes** |
| **Propósito** | Permite gestionar los Pacientes. |
| **Descripción** | Permite gestionar los Pacientes nuevos o asegurados de la clínica. |
| **Actores** | Personal Administrativo, Administrador |
| **Actor Inicial** | Personal Administrativo |
| **Pre-Condición** | Ninguna |
| **Proceso** | **Insertar Paciente:**  1.1. Ingresar la información del paciente.  1.2. Insertar los datos.  1.3. Guardar los datos.  **Modificar Paciente:**  2.1. Seleccionar al paciente.  2.2. Modificar los datos.  2.3. Guardar los cambios.  **Eliminar Paciente:**  3.1. Seleccionar al paciente.  3.2. Confirmar la eliminación.  3.3. Guardar los cambios. |
| **Postcondición** | Ninguna |
| **Excepción** | 1. Datos incorrectos  2. No se seleccionó ninguna paciente  3. No se confirmó la eliminación |

### **5.4.3. CU03 Gestionar Emergencias**

*Ilustración 15: Gestionar Emergencias*

*Fuente (Elaboración Propia)*

|  |  |
| --- | --- |
| **Casos de Uso** | **CU03: Gestionar Emergencias** |
| **Propósito** | Permite Gestionar las Emergencias. |
| **Descripción** | Permite gestionar las emergencias recepcionadas así mismo pudiendo modificar o visualizar información. |
| **Actores** | Personal Médico, Personal Administrativo |
| **Actor Inicial** | Personal Administrativo |
| **Pre-Condición** | Debe estar en estar registrado como paciente |
| **Proceso** | **Visualizar Emergencia:**  1.1. Seleccionar la emergencia.  1.2. Ingresar los detalles de la emergencia.  1.3. Visualizar Información.  **Modificar Emergencia:**  2.1. Seleccionar la emergencia a modificar.  2.2. Modificar los detalles de la emergencia.  2.3. Guardar los cambios. |
| **Postcondición** | Ninguna |
| **Excepción** | 1. Datos incorrectos  2. No se seleccionó ninguna paciente  3. No se confirmó la eliminación |

### **5.4.4. CU04 Identificación de Pacientes**

  
*Ilustración 16: Identificación de Pacientes*

*Fuente (Elaboración Propia)*

|  |  |
| --- | --- |
| **Casos de Uso** | **CU04: Identificación de Pacientes** |
| **Propósito** | Identificar al paciente mediante una foto y utilizar la detección facial para recuperar e imprimir la información más relevante de su historial médico. En caso de no encontrar una coincidencia, registrar al paciente como usuario nuevo. |
| **Descripción** | El personal médico puede realizar las acciones de identificación y ver los registros devueltos por el mismo. |
| **Actores** | Personal Médico, Administrador |
| **Actor Inicial** | Personal Médico |
| **Pre-Condición** | Ninguna |
| **Proceso** | 1. Tap en la opción de Identificación de Paciente.  2. Seleccionar Identificación de Paciente  3. Tomar y Subir Fotografía  4. Registrar actividad de paciente en el área de emergencia. |
| **Postcondición** | Registro de actividad del paciente en área de emergencia. |
| **Excepción** | Para registrar un paciente, se debe tomar en cuenta que sea nuevo y no uno ya asegurado o registrado. |

### **5.4.5. CU05 Designación de Análisis**

  
*Ilustración 17: Designación de Análisis*

*Fuente (Elaboración Propia)*

|  |  |
| --- | --- |
| **Casos de Uso** | **CU05: Designación de Análisis** |
| **Propósito** | Adicionar información sobre análisis en caso de que el paciente lo requiera. |
| **Descripción** | El personal médico puede adicionar análisis requeridos para el paciente. |
| **Actores** | Personal Médico, Administrador |
| **Actor Inicial** | Personal Médico |
| **Pre-Condición** | Ninguna |
| **Proceso** | 1. Tap en la opción de la Emergencia  2. Seleccionar Adicionar Análisis  3. Registrar datos de análisis requerido.  4. Guardar información del análisis requerido para paciente en el área de emergencia. |
| **Postcondición** | Registro de información de análisis requeridos para paciente en área de emergencia. |
| **Excepción** | Error en los datos introducidos. |

### **5.4.6. CU06 Gestionar Historia Médica**

*Ilustración 18: Gestionar Historia Médica*

*Fuente (Elaboración Propia)*

|  |  |
| --- | --- |
| **Casos de Uso** | **CU06: Gestionar Historia Médica** |
| **Propósito** | Registrar, modificar y eliminar información de la historia médica. |
| **Descripción** | El personal médico puede adicionar informacional adicional o actualizar la información base de la historia médica de un paciente. |
| **Actores** | Personal Médico, Administrador |
| **Actor Inicial** | Personal Médico |
| **Pre-Condición** | Ninguna |
| **Proceso** | \*Dar un Tap en el Módulo de Historia Médica 1) Registrar   1. Dar clic en el botón Nueva Historia Medica 2. Asocia el paciente y info a la Historia Médica 3. Dar clic en el botón guardar   2) Modificar   1. Dar clic en el botón Actualizar Historia Médica 2. Actualiza nuevos datos en la Historia Médica. 3. Dar clic en el botón guardar |
| **Postcondición** | Registro de actividad del paciente en área de emergencia. |
| **Excepción** | Para actualizar algún dato, la historia médica ya debe estar creada y asociada a un paciente, si no existe se debe registrar una nueva. |

## **5.5. Diagrama General de Casos de Uso**

*Ilustración 19: Diagrama General de Casos de Uso*

*Fuente (Elaboración Propia)*

# **CAPÍTULO VI**

**ANÁLISIS**

## **6.1. Análisis de Paquetes**

### **6.1.1. Identificación de Paquetes**

*Ilustración 20: Identificación de Paquetes*

*Fuente (Elaboración Propia)*

## **6.2. Análisis de Arquitectura**

### **6.2.1. Vista de Paquetes** Gestión de Usuario

*Ilustración 21: Vista de Paquete desde Gestión de Usuario*

*Fuente (Elaboración Propia)*

  
Gestión de Emergencia

*Ilustración 22: Vista de Paquete desde Gestión de Emergencia*

*Fuente (Elaboración Propia)*

Gestión de Historia Médica  
 *Ilustración 23: Vista de Paquete desde Gestión de Historia Médica*

*Fuente (Elaboración Propia)*

## **6.3. Análisis de Casos de Uso**

### **6.3.1. Diagrama de Comunicación**

CU01: Gestionar Personal Médico

*Ilustración 24: Diagrama de Comunicación de Gestionar Personal Médico*

*Fuente (Elaboración Propia)*

CU02: Gestionar Pacientes

*Ilustración 25: Diagrama de Comunicación de Gestionar Personal Médico*

*Fuente (Elaboración Propia)*

CU03: Gestionar Emergencias

*Ilustración 26: Diagrama de Comunicación de Gestionar Emergencias*

*Fuente (Elaboración Propia)*

CU04: Identificación de Pacientes

*Ilustración 27: Diagrama de Comunicación de Identificación de Pacientes*

*Fuente (Elaboración Propia)*

CU05: Designación de Análisis

*Ilustración 28: Diagrama de Comunicación de Designación de Análisis  
Fuente (Elaboración Propia)*

CU06: Gestionar Historia Médica

*Ilustración 29: Diagrama de Comunicación de Designación de Análisis  
Fuente (Elaboración Propia)*

# **CAPÍTULO VII**

**ARQUITECTURA**

## **7.1. Diseño de la Arquitectura**

**7.1.1. Diseño de la Arquitectura Física**

*Ilustración 30: Diseño de la Arquitectura Física*

*Fuente (Elaboración Propia)*

## **7.2. Diagrama de Secuencia**

CU01: Gestionar Personal Médico

*Ilustración 31: Diagrama de Secuencia de Gestionar Personal Médico*

*Fuente (Elaboración Propia)*

CU02: Gestionar Pacientes

*Ilustración 32: Diagrama de Secuencia de Gestionar Pacientes*

*Fuente (Elaboración Propia)*

CU03: Gestionar Emergencias



*Ilustración 33: Diagrama de Secuencia de Gestionar Emergencias*

*Fuente (Elaboración Propia)*

CU04: Identificación de Pacientes



*Ilustración 34: Diagrama de Secuencia de Identificación de Pacientes*

*Fuente (Elaboración Propia)*

CU05: Designación de Análisis



*Ilustración 35: Diagrama de Secuencia de Designación de Análisis*

*Fuente (Elaboración Propia)*

CU06: Gestionar Historia Médica

*Ilustración 36: Diagrama de Secuencia de Gestionar Historia Médica*

*Fuente (Elaboración Propia)*

## **Diseño de Datos**

### **Modelo de Dominio**



### **7.3.2. Diseño Conceptual**

*Ilustración 37: Diseño Conceptual de Base de Datos*

*Fuente (Elaboración Propia)*

### **7.3.3. Diseño Lógico**

Usuario

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **carnet** | **nombre** | **email** | **sexo** | **role** | **grupo** | **telefono** | **fecha\_nac** | **tipo\_sangre** | **created\_at** | **updated\_at** |
| PK |  |  |  |  |  |  |  |  |  |  |  |

Personal Médico

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **especialidad** | **usuario\_id** | **created\_at** | **updated\_at** |
| **PK** |  | **FK** |  |  |

Paciente

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **id** | **contacto\_emerg** | **usuario\_id** | **created\_at** | **updated\_at** |
| **PK** |  | **FK** |  |  |

Emergencia

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **motivo** | **gravedad** | **observacion** | **estado** | **fecha** | **hora** | **diagnostico** | **detalle\_fin** | **paciente\_id** | **medico\_id** | **created\_at** | **updated\_at** |
| **PK** |  |  |  |  |  |  |  |  | **FK** | **FK** |  |  |

Analisis

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **id** | **descripcion** | **motivo** | **fecha** | **emergencia\_id** | **created\_at** | **updated\_at** |
| **PK** |  |  |  | **FK** |  |  |

Historia Médica

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **id** | **antemed\_fam** | **antemed\_pers** | **medis\_act** | **alergias** | **h\_enfermedades** | **h\_cirugias** | **salud\_actual** | **notas** | **paciente\_id** | **created\_at** | **updated\_at** |
| **PK** |  |  |  |  |  |  |  |  | **FK** |  |  |

### **7.3.4. Diseño Físico**

CREATE DATABASE melendres\_bd;

CREATE TABLE Usuario (

id INT PRIMARY KEY,

carnet VARCHAR(255),

nombre VARCHAR(255),

email VARCHAR(255),

sexo VARCHAR(255),

role VARCHAR(255),

group VARCHAR(255),

telefono VARCHAR(255),

fecha\_nac VARCHAR(255),

tipo\_sangre VARCHAR(255),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Medico (

id INT PRIMARY KEY,

especialidad VARCHAR(255),

FOREIGN KEY (id) REFERENCES Usuario(id),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Paciente (

id INT PRIMARY KEY,

contacto\_emerg VARCHAR(255),

FOREIGN KEY (id) REFERENCES Usuario(id),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Emergencia (

id INT PRIMARY KEY,

motivo VARCHAR(255),

gravedad VARCHAR(255),

observacion VARCHAR(255),

estado VARCHAR(255),

fecha DATE,

hora TIME,

diagnostico VARCHAR(255),

detalle\_fin VARCHAR(255),

paciente\_id INT,

medico\_id INT,

FOREIGN KEY (paciente\_id) REFERENCES Paciente(id),

FOREIGN KEY (medico\_id) REFERENCES Medico(id),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE Analisis (

id INT PRIMARY KEY,

descripcion VARCHAR(255),

motivo VARCHAR(255),

fecha DATE,

emergencia\_id INT,

FOREIGN KEY (emergencia\_id) REFERENCES Emergencia(id),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

CREATE TABLE "Historia Medica" (

id INT PRIMARY KEY,

antemed\_fam VARCHAR(255),

antemed\_pers VARCHAR(255),

medis\_act VARCHAR(255),

alergias VARCHAR(255),

h\_enfermedades VARCHAR(255),

h\_cirugias VARCHAR(255),

salud\_actual VARCHAR(255),

notas VARCHAR(255),

paciente\_id INT,

FOREIGN KEY (paciente\_id) REFERENCES Paciente(id),

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

updated\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

-- Restricciones de integridad referencial

ALTER TABLE Emergencia ADD CONSTRAINT fk\_emergencia\_paciente

FOREIGN KEY (paciente\_id) REFERENCES Paciente(id);

ALTER TABLE Emergencia ADD CONSTRAINT fk\_emergencia\_medico

FOREIGN KEY (medico\_id) REFERENCES Medico(id);

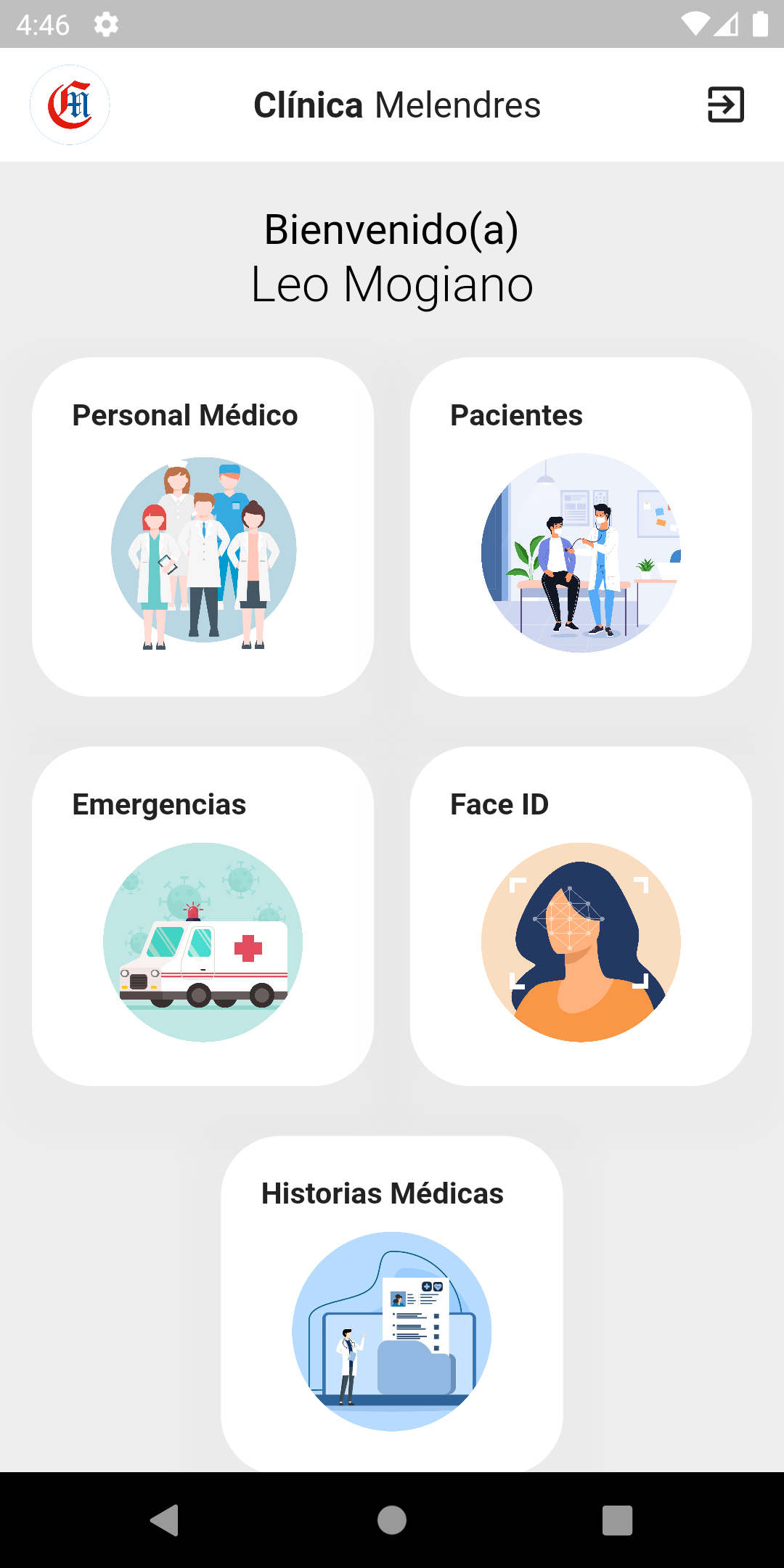
ALTER TABLE Analisis ADD CONSTRAINT fk\_analisis\_emergencia

FOREIGN KEY (emergencia\_id) REFERENCES Emergencia(id);

ALTER TABLE "Historia Medica" ADD CONSTRAINT fk\_historia\_paciente

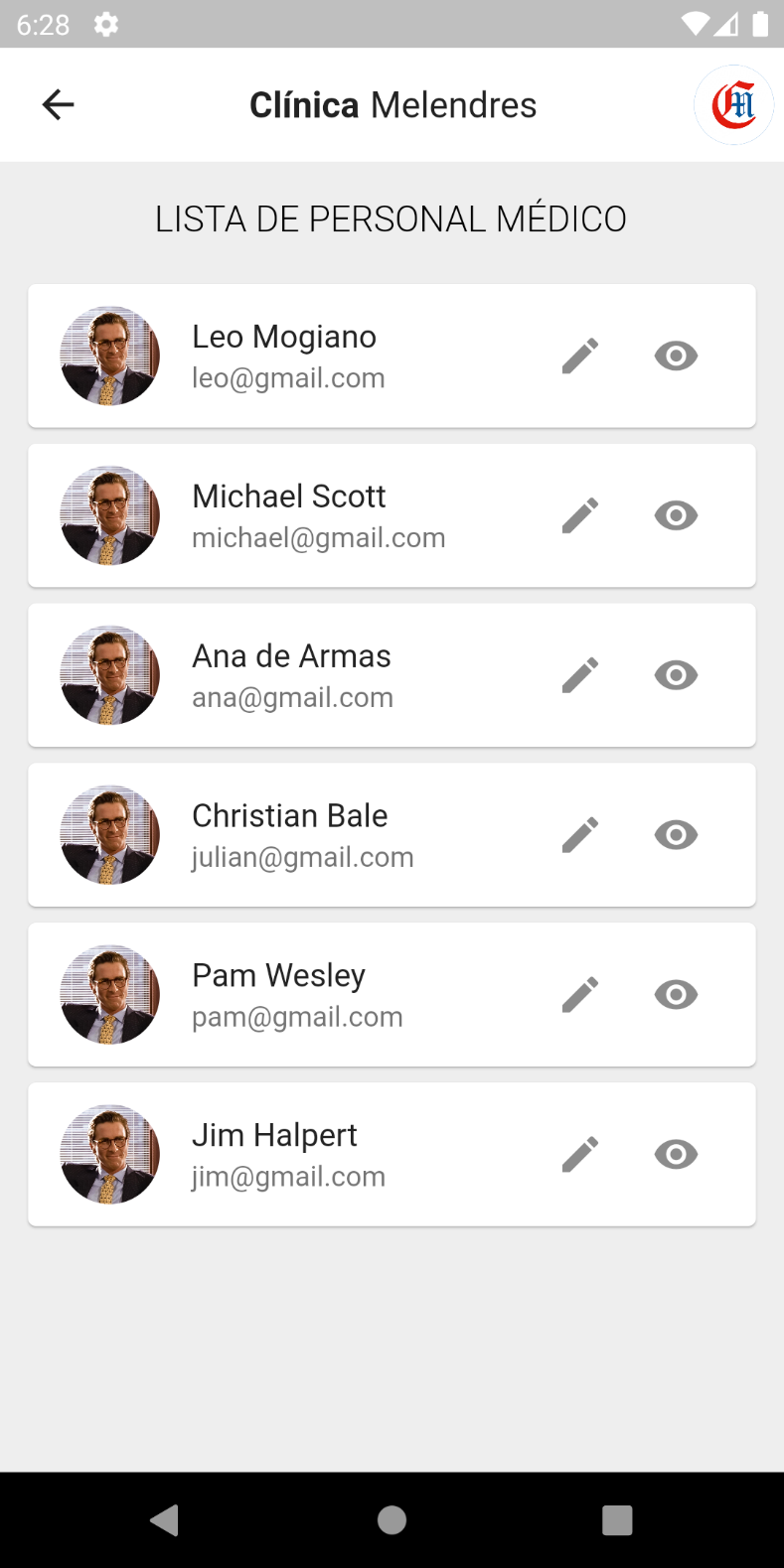
FOREIGN KEY (paciente\_id) REFERENCES Paciente(id);

## **7.4. Diseño de Interfaz**

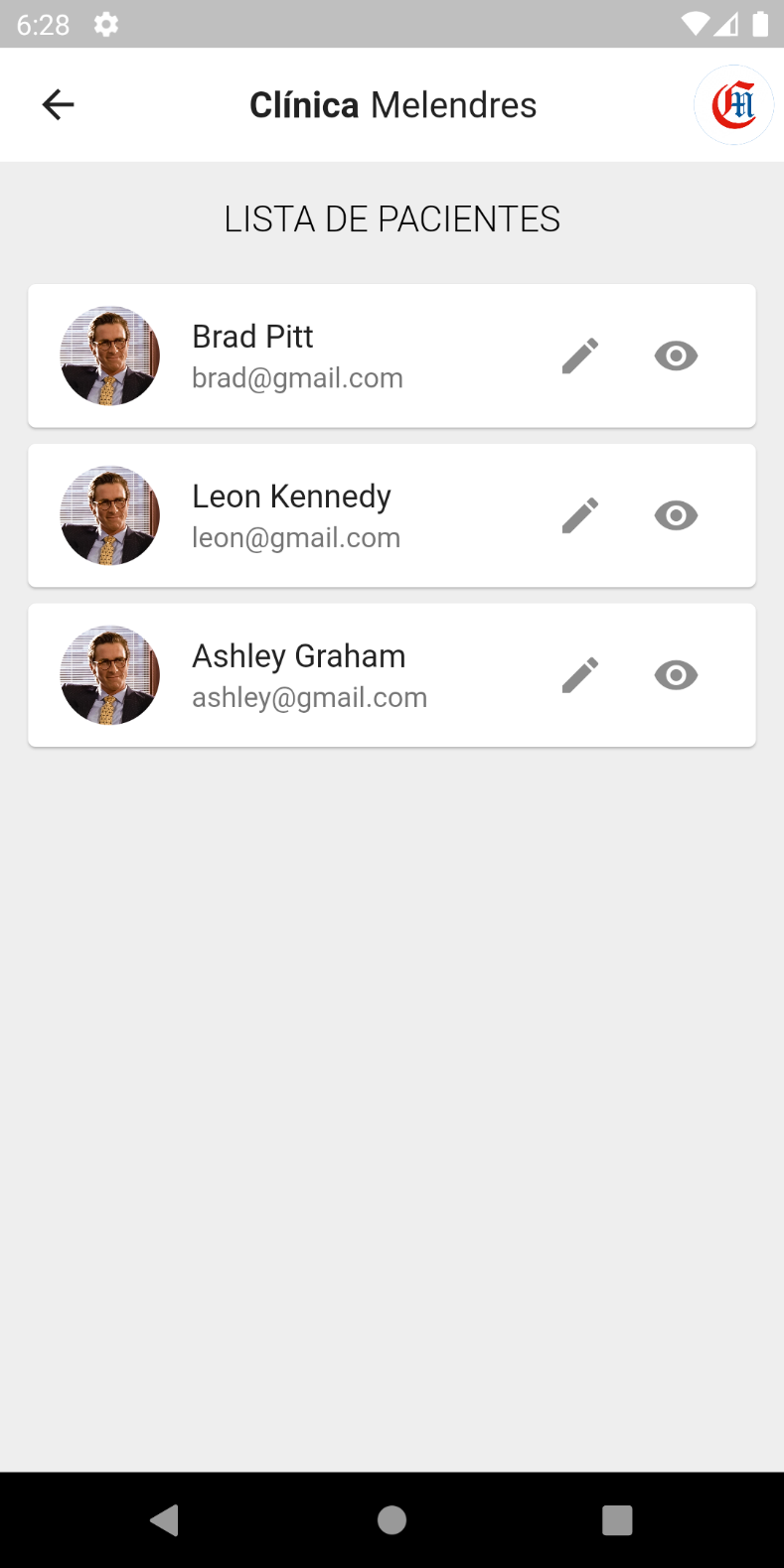


*Ilustración 38: Diseño de Pantalla Inicial*

*Fuente (Elaboración Propia)*

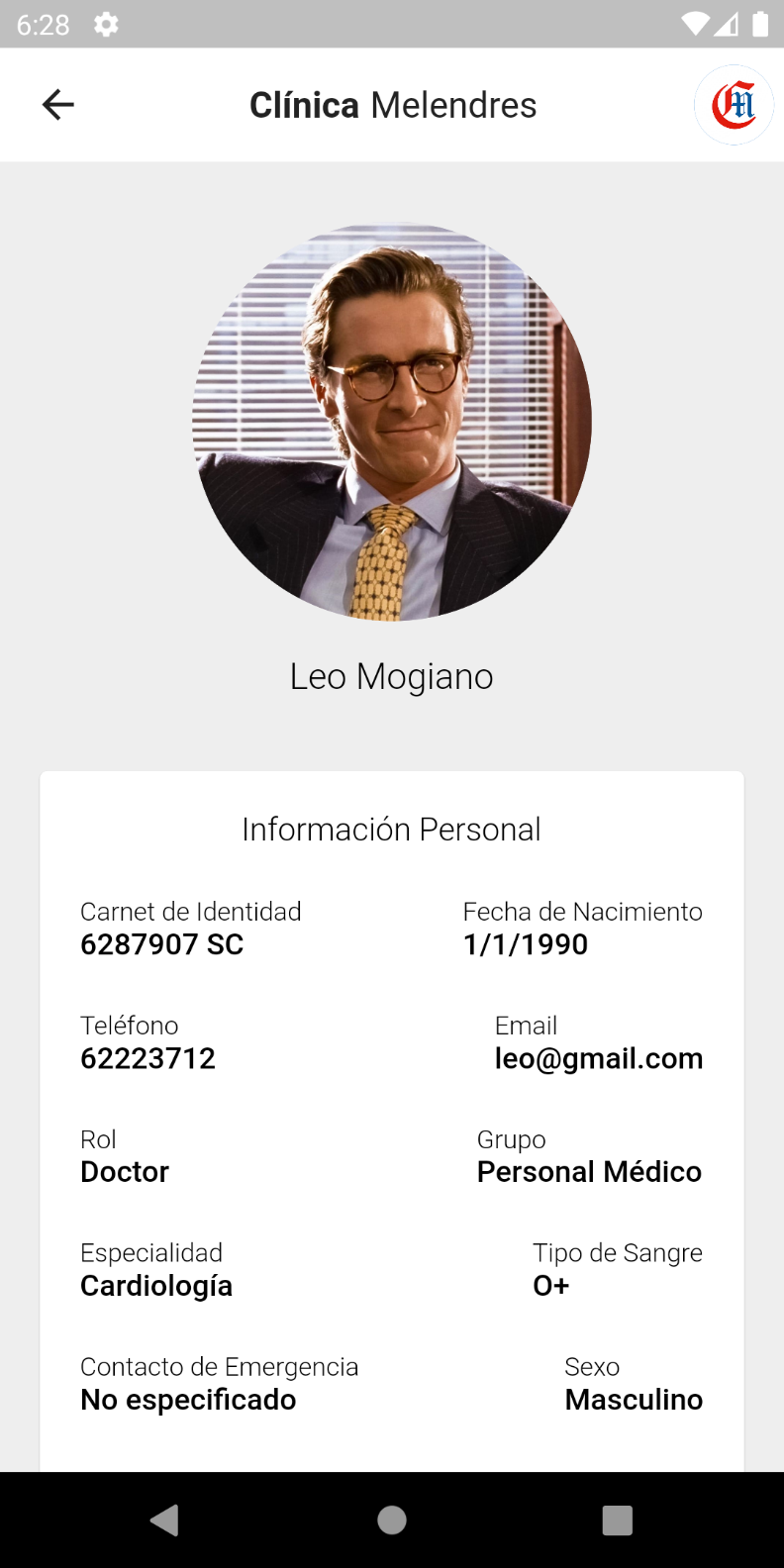
  
  
*Ilustración 39: Diseño de Gestionar Personal Médico*

*Fuente (Elaboración Propia)*



*Ilustración 40: Diseño de Gestionar Pacientes*

*Fuente (Elaboración Propia)*



*Ilustración 41: Diseño de Gestionar Pacientes*

*Fuente (Elaboración Propia)*

*****Ilustración 42: Diseño de Gestionar Pacientes*

*Fuente (Elaboración Propia)*

# **CAPÍTULO VIII**

**IMPLEMENTACIÓN**

## **8.1. Implementación de la Arquitectura**

### **8.1.1. Implementación de Componentes**



*Ilustración 43: Implementación de Componentes*

*Fuente (Elaboración Propia)*

## **8.2. Herramientas de Desarrollo**

### **8.2.1. Sistema Operativo**

La plataforma que soporta el software desarrollado es el S.O “Windows 10”. Se eligió esta plataforma porque ofrece todas las herramientas necesarias para su implementación y al ser uno de los sistemas operativos más usados en el mundo.

Ventajas:

* + Es más conocido.
  + Es el que tiene más software desarrollado.
  + Se puede aprender desde cómo usar el Ratón hasta saber lo que es un Firewall.
  + Tiene una gran cantidad de tutoriales.
  + No viene con CD de instalación o de recuperación. El Starter posee una imagen en el disco rígido de la nueva computadora.
  + Incluye los programas clásicos y habituales de Windows en versiones integras, pero limitadas de acuerdo con las prestaciones anteriormente reseñadas: Paint, Wordpad, Internet Explorer, Outlook Express, Windows Media Player, Service Pack.

Desventajas:

* Las nuevas versiones requieren muchos recursos.
* La mayoría de los virus están hechos para Windows.
* Puedes tener errores de compatibilidad en sistemas nuevos.
* Limitaciones de actualización de Hardware.

### **8.2.2. Lenguaje de Programación**

PHP

PHP es un lenguaje popular para crear APIs. Es versátil y cuenta con numerosas herramientas y bibliotecas para desarrollar APIs eficientes y escalables. Su sintaxis sencilla y amplia adopción lo convierten en una opción accesible para crear APIs personalizadas y potentes.

DART

Dart es un lenguaje de programación enfocado en el desarrollo de aplicaciones móviles. Con su enfoque en la creación de interfaces de usuario y su capacidad de desarrollo multiplataforma, Dart permite a los desarrolladores crear aplicaciones móviles eficientes y de alto rendimiento con facilidad.

### **8.2.3. Base de Datos**

PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos relacional de código abierto y altamente escalable. Se basa en el lenguaje de consulta estructurado (SQL) y está respaldado por una comunidad activa de desarrolladores.

Una de las principales fortalezas de PostgreSQL es su capacidad para gestionar grandes volúmenes de datos y manejar cargas de trabajo intensivas. Proporciona una amplia gama de características avanzadas, como transacciones ACID (Atomicidad, Consistencia, Aislamiento y Durabilidad), integridad referencial, consultas complejas y replicación.

PostgreSQL es conocido por su capacidad de extensibilidad, lo que permite a los desarrolladores personalizar y ampliar sus funcionalidades según sus necesidades. También es compatible con una variedad de lenguajes de programación y frameworks, lo que facilita su integración en proyectos de desarrollo de aplicaciones móviles, web y empresariales.

### **8.2.4. Herramientas Case**

Enterprise Architect

Enterprise Architect es una poderosa herramienta de modelado y diseño de software diseñado para ayudar en la gestión y desarrollo de proyectos de software a gran escala. Con una amplia gama de características y capacidades, Enterprise Architect es ampliamente utilizado en entornos empresariales y de desarrollo de software.

Asimismo, permite a los equipos de desarrollo visualizar, analizar y diseñar arquitecturas de software complejas. Proporciona una interfaz intuitiva y fácil de usar que permite a los usuarios crear diagramas de casos de uso, diagramas de clase, diagramas de secuencia, diagramas de componentes y muchos otros tipos de diagramas utilizados en la ingeniería de software.

# **CAPÍTULO IX**

**COSTOS, RIESGOS Y SEGURIDAD**

## **9.1. Costos**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Fecha** | **Fecha** |  | **Pre.** | **% Deprec** | **Prec** | **Precio total** |
| **Recurso** | **Inicio** | **Fin** | **Cantidad** | **Unit** | **Unit** | **(Bs)** |
|  |  |  |  | **(Bs)** | **neto** |  |
| **Hardware** | | | | | | | | |
| Portátil HP Ryzen 3 | 13/4/2023 | 28/7/2023 | 1 | 5000 | 8.3% | 4585 | 4585 |
| Portátil HP | 13/4/2023 | 28/7/2023 | 1 | 7000 | 8.3% | 6419 | 6419 |
| iCore7 12th |
| Router | 13/4/2023 | 28/7/2023 | 2 | 400 | 8.3% | 367 | 734 |
| Mouse Microsoft | 13/4/2023 | 28/7/2023 | 2 | 120 | 8.3% | 110 | 220 |
| Teclado Lexus | 13/4/2023 | 28/7/2023 | 2 | 100 | 8.3% | 92 | 184 |
| Monitor Samsung | 13/4/2023 | 28/7/2023 | 2 | 1200 | 8.3% | 1100 | 2200 |
| Repetidores TP Link | 13/4/2023 | 28/7/2023 | 4 | 350 | 8.3% | 321 | 1284 |
| **Software** | | | | | | | | |
| Microsoft Office 365 | 13/4/2023 | 28/7/2023 | 2 | 500 | - | 500 | 1000 |
| Enterprise Architect | 13/4/2023 | 28/7/2023 | 2 | 1500 | - | 1500 | 3000 |
| Zoom | 13/4/2023 | 28/7/2023 | 2 | 1050 | - | 1050 | 2100 |
| AWS | 13/4/2023 | 28/7/2023 | 1 | 500 | - | 500 | 500 |
| **Servicios /infraestructura** | | | | | | | | |
| Luz | 13/4/2023 | 28/7/2023 | 2 | 200 | - | 200 | 400 |
| Agua | 13/4/2023 | 28/7/2023 | 2 | 150 | - | 150 | 300 |
| Internet | 13/4/2023 | 28/7/2023 | 2 | 400 | - | 400 | 800 |
| Escritorio | 13/4/2023 | 28/7/2023 | 2 | 260 | 4.3% | 249 | 498 |
| Silla giratoria | 13/4/2023 | 28/7/2023 | 2 | 200 | 4.3% | 191 | 382 |
| Impresora | 13/4/2023 | 28/7/2023 | 2 | 600 | 4.3% | 574 | 1148 |
| **Persona** |  |  |  |  |  |  |  |
| Programador | 13/4/2023 | 28/7/2023 | 2 | 6000 | - | 6000 | 12000 |
| Programas otros | 13/4/2023 | 28/7/2023 | 2 | 6500 | - | 6500 | 13000 |
| Total |  |  |  |  |  |  | 50754 |

## **9.2. Riesgos**

### **9.2.1. Identificación de Riesgos**

1. Abandono de un miembro del equipo (ya sea de desarrollo en implementación o artefactos)
2. Estimación o planificación del tiempo errónea
3. Insuficiente documentación de un componente
4. Error de diseño por falta de coordinación
5. Incompatibilidad o complicación de alguna tecnología externa con lo planeado para el desarrollo del proyecto
6. Cambios en el diseño no planificados
7. Incapacidad de un miembro del equipo
8. Balanceo de cargas muy desequilibrado en el equipo de desarrollo
9. Inexperiencia en algún ámbito del proyecto propuesto
10. Falta de planificación con las presentaciones

### **9.2.2. Plan de Aversión**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Plan de aversión** | | | | |
| **Riesgo** | **Probabilidad De Presencia** | **Escala**  **Impacto** | **Estrategias para reducir la probabilidad de ocurrencia** | **Estrategias para reducir el impacto** |
| **Abandono de un miembro del equipo (ya sea de desarrollo en implementación o artefactos)** | 60% | **MODERADO** | * Establecer un pacto formal * Flexibilidad en la asignación de tareas para apoyarse | Seguir un plan de trabajo flexible y rápido |
| **Estimación o planificación del tiempo errónea** | 40% | **CRITICO** | * Seguir una metodología de desarrollo completa para la determinación de estimaciones * Realizar las estimaciones en grupo | Más atención en la resolución de la planificación. |
| **Insuficiente documentación de un componente** | 20% | **SIGNIFICATIVO** | Mejorar en los diagramas | Realizar reuniones de review cada cierto tiempo |
| **Error de diseño por falta de coordinación** | 40% | **MODERADO** | Realizar reuniones de coordinación | Simplificar los artefactos para que el objetivo de este sea fácil de identificar |
| **Incompatibilidad o complicación de alguna tecnología externa con lo planeado para el desarrollo del proyecto** | 10% | **MODERADO** | Investigar moderadamente las tecnologías a usar | Usar tecnologías populares y con comunidades grandes de la misma |
| **Cambios en el diseño no planificados** | 50% | **SIGNIFICATIVO** | Realizar una documentación moderadamente detallada | Usar una metodología iterativa |
| **Incapacidad de un miembro del equipo** | 30% | **SIGNIFICATIVO** | Realizar reuniones diarias para saber el estado de cada miembro en su avance | Asignar las tareas más complejas propensas a este riesgo en equipo |
| **Balanceo de cargas muy desequilibrado en el equipo de desarrollo** | 10% | **MODERADO** | Realizar reuniones con antelación para que todos los miembros del equipo estén presentes | Dividir el equipo en sub-grupos |
| **Inexperiencia en algún ámbito del proyecto propuesto** | 40% | **MODERADO** | Investigar sobre el tema propuesto  Contactar a un tercero cercano del equipo con experiencia en el tema | Realizar un banco de preguntas con los temas que se tocaran en el desarrollo del proyecto |
| **Falta de planificación con las presentaciones** | 20% | **SIGNIFICATIVO** | Tener un calendario y dashboard como en herramientas como jira o google calendar | Gestionar mejor las ideas y también usar |

## **9.3. Seguridad**

### **9.3.1. Seguridad Física**

Dentro de las amenazas físicas podemos englobar cualquier error o daño en el hardware que se puede presentar en cualquier momento. Por ejemplo, daños en discos duros, en los procesadores, errores de funcionamiento de la memoria, etc. Todos ellos hacen que la información no esté accesible o no sea fiable. Otro tipo de amenazas físicas son las catástrofes naturales. Por ejemplo, hay zonas geográficas del planeta en las que las probabilidades de sufrir terremotos, huracanes, inundaciones, etc. son mucho más elevadas. En estos casos en los que es la propia Naturaleza la que ha provocado el desastre de seguridad, no por ello hay que descuidarlo e intentar prever al máximo este tipo de situaciones. (Shibuya, 2019).

**Riesgos Físicos**

* Desastres naturales
* Incendios
* Sobrecalentamiento
* Mala ventilación
* Mala gestión de energía

### **9.3.2. Seguridad Lógica**

Si bien las amenazas físicas pueden incluir robo, vandalismo y daño ambiental, las amenazas lógicas son aquellas que pueden dañar los sistemas de software, datos o red sin dañar realmente el hardware.

Los virus, los ataques DoS, el phishing y los troyanos son ejemplos de amenazas lógicas que se pueden descargar o propagar inadvertidamente mediante el uso de unidades flash. En el peor de los casos, un ataque no solo afecta a un solo dispositivo, sino que se propaga a través de una red, raspando datos confidenciales o apagando los sistemas operativos. (León, 2022)

**Riesgos Lógicos**

* Malware
* Programas no testeados
* Errores del usuario
* Mal uso de la tecnología

# **CAPÍTULO X**

**PRUEBAS**

## **10.1. Planificación de Pruebas**

Las pruebas tienen como objetivo la comprobación de la fiabilidad y estabilidad del sistema. De esta manera se pueden identificar la mayoría de los errores posibles en los procesos más utilizados, pudiendo así corregirlos y hacer que el sistema sea lo más eficaz y eficiente posible.

**Pruebas de Caja Blanca**

Las pruebas de caja blanca son pruebas estructurales, en las que conociendo el código y siguiendo su estructura lógica, están destinadas a comprobar que el código hace correctamente lo que el diseño de bajo nivel indica o demostrar que no se comporta adecuadamente ante determinadas situaciones. Estas pruebas se han ido realizando a la que vez que se iba desarrollando el código, de forma que cada módulo ha sido probado de forma independiente al resto.

**Pruebas de Caja Negra**

Son pruebas funcionales. Se parte de los requisitos funcionales, a muy alto nivel, para diseñar pruebas que se aplican sobre el sistema sin necesidad de conocer como está construido por dentro (Caja negra). Las pruebas se aplican sobre el sistema empleando un determinado conjunto de datos de entrada y observando las salidas que se producen para determinar si la función se está desempeñando correctamente por el sistema bajo prueba. Las herramientas básicas son observar la funcionalidad y contrastar con la especificación.

## **10.2. Diseño de Pruebas**

### **10.2.1. Identificación y Estructuración de Procedimiento de Prueba**

Para el procedimiento de pruebas, nos centraremos en las pruebas de la caja negra, de acuerdo con la siguiente estructura:

* **Nombre:** Nombre completo del caso de uso a realizar la prueba.
* **Descripción:** Una breve descripción del caso de uso a tratar.
* **Precondición:** Condición necesaria que debe cumplirse antes de ejecutar el caso de prueba.
* **Resultado Esperado**: Es lo que se espera al concluir la prueba, éste puede fracasar o terminar con éxito.
* **Resultado Obtenido:** Es el resultado real obtenido al finalizar la prueba.

## **10.3. Pruebas Realizadas**

**Pruebas de Caja Negra**

CU01 Gestionar Personal Médico

|  |  |
| --- | --- |
| Nombre | Prueba de Gestionar Personal Médico |
| Descripción | Permite gestionar al personal médico, doctores y enfermeras, registrando, modificando y eliminando sus datos dentro del software |
| Precondición | Debe iniciar sesión |
| Datos de Entrada | Datos del personal medico |
| Captura de Imagen Inicial |  |
| Pasos | 1. En la pantalla menú presionar el botón Personal medico 2. Presionar el botón circular con el signo de más en la parte inferior derecha 3. Rellenar los datos    1. Mensajes de error en los campos no rellenado esenciales    2. Mensaje de error de formato si no tiene el formato de correo electrónico correcto 4. Presionar el botón guardar   en la lista de personal médico se puede modificar y ver los datos de cada personal medico |
| Resultado Esperado | El usuario puede rellanar sus datos correctamente de manera sencilla |
| Resultado Obtenido | Se accede a la sección y el usuario puede rellenar los datos |
| Captura de Imagen Final |  |
| Resultado de Prueba | CORRECTO |

CU02 Gestionar Pacientes

|  |  |
| --- | --- |
| Nombre | Prueba de Gestionar Pacientes |
| Descripción | Permite gestionar los Pacientes nuevos o asegurados de la clínica. |
| Precondición | Debe iniciar sesión |
| Datos de Entrada | Datos del nuevo paciente |
| Captura de Imagen Inicial |  |
| Pasos | 1. En la pantalla menú presionar el botón Paciente 2. Presionar el botón circular con el signo de más en la parte inferior derecha 3. Rellenar los datos    1. Mensajes de error en los campos no rellenado esenciales    2. Mensaje de error de formato si no tiene el formato de correo electrónico correcto 4. Presionar el botón guardar   en la lista de pacientes se puede modificar y ver los datos de cada paciente nuevo |
| Resultado Esperado | El usuario puede rellanar sus datos correctamente de manera sencilla |
| Resultado Obtenido | Se accede a la sección y el usuario puede rellenar los datos |
| Captura de Imagen Final |  |
| Resultado de Prueba | CORRECTO |

CU03 Gestionar Emergencias

|  |  |
| --- | --- |
| Nombre | Prueba de Gestionar Emergencias |
| Descripción | Permite gestionar las emergencias recepcionadas así mismo pudiendo modificar o visualizar información. |
| Precondición | Debe iniciar sesión |
| Datos de Entrada | Datos sobre la emergencia del paciente |
| Captura de Imagen Inicial |  |
| Pasos | 1. En la pantalla menú presionar el botón Emergencia 2. Presionar el botón circular con el signo de más en la parte inferior derecha 3. Rellenar los datos    * Seleccionar gravedad de la emergencia    * Mensajes de error en los campos no rellenado esenciales 4. Presionar el botón guardar   en la lista de Emergencia se puede modificar y ver los datos de cada paciente nuevo |
| Resultado Esperado | El usuario puede rellanar sus datos correctamente de manera sencilla |
| Resultado Obtenido | Se accede a la sección y el usuario puede rellenar los datos |
| Captura de Imagen Final |  |
| Resultado de Prueba | CORRECTO |

CU04 Identificación de Pacientes

|  |  |
| --- | --- |
| Nombre | Prueba de Identificación de Pacientes |
| Descripción | El personal médico puede realizar las acciones de identificación del paciente y ver los registros devueltos por el mismo. |
| Precondición | Debe iniciar sesión, debe ser personal médico o administrativo |
| Datos de Entrada | imágenes .jpg .gif .png |
| Captura de Imagen Inicial |  |
| pasos | 1. En la pantalla menú presionar el botón Identificación Facial 2. Presionar el botón circular con el signo de más en la parte inferior derecha 3. Tomar fotografía o subir Fotografía    1. Si es primera vez que utiliza la cámara dentro de la app o primera vez que sube una fotografía deberá permitir el acceso 4. Mensajes de paciente reconocido, con su nombre o si no lo reconoce paciente no encontrado, y lo redirecciona al crear paciente 5. Presionar el botón guardar   en la lista de Emergencia se puede modificar y ver los datos de cada paciente nuevo |
| Resultado Esperado | El usuario puede tomar una fotografía o subir una fotografía correctamente de manera sencilla |
| Resultado Obtenido | Se accede a la sección y el usuario tomar una fotografía o subir una fotografía |
| Captura de Imagen Final |  |
| Resultado de Prueba | CORRECTO |

CU5 Gestionar Historia Médica

|  |  |
| --- | --- |
| Nombre | Prueba de Gestionar Historia Médica |
| Descripción | El personal médico puede adicionar informacional adicional o actualizar la información base de la historia médica de un paciente. |
| Precondición | Debe iniciar sesión |
| Datos de Entrada | Ninguno |
| Captura de Imagen Inicial |  |
| Pasos | 1. En la pantalla menú presionar el botón Historia medica 2. En la lista de Emergencia se puede modificar y ver los datos de cada paciente nuevo |
| Resultado Esperado | El usuario podrá ver todas las historias medicas de los pacientes |
| Resultado Obtenido | Se logro visualizar las historias médicas y ver a detalle todas las especificaciones de esta |
| Captura de Imagen Final |  |
| Resultado de Prueba | CORRECTO |

# **CONCLUSIÓN**

En el campo de la medicina, la tecnología juega un papel crucial. La solución que queremos dar para la clínica Melendres, una aplicación móvil con reconocimiento facial demuestra cómo la tecnología transforma y mejora los procesos médicos.

Utilizando inteligencia artificial y redes neuronales convolucionales, se creó una solución innovadora que agiliza la identificación de pacientes en emergencias. El reconocimiento facial en la aplicación permite al personal médico acceder rápidamente a información relevante.

La tecnología en la medicina puede ayudar a optimizar los tiempos en las actividades que realizan como área de emergencia, donde cada segunda cuenta. Además, la tecnología mejora la calidad de las decisiones de los médicos y la efectividad de los tratamientos.

En conclusión, el proyecto para la clínica Melendres destaca el impacto positivo y la importancia de la tecnología en la medicina. La implementación de la aplicación móvil con reconocimiento facial mejora la eficiencia en la identificación de pacientes en emergencias, brindando una atención más rápida y precisa. Esta integración tecnológica es esencial para garantizar una atención médica de calidad, salvar vidas y avanzar en el conocimiento científico en beneficio de la sociedad.

# BIBLIOGRAFÍA

* Jain, A. K., & Li, S. Z. (2011). Handbook of face recognition (2nd Ed.). Springer.
* Turk, M., & Pentland, A. (1991). Face recognition using eigenfaces. In Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 586-591). IEEE.
* Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001 (Vol. 1, pp. I-511-I-518). IEEE.
* Yang, M., Kriegman, D., & Ahuja, N. (2002). Detecting faces in images: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(1), 34-58.
* Zhao, W., Chellappa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. ACM Computing Surveys (CSUR), 35(4), 399-458.
* Kadir, A. A., & Kamaruddin, L. M. (2013). A comparative study between LBP and Haar-like features for face detection. In 2013 International Conference on Advanced Computer Science Applications and Technologies (pp. 57-62). IEEE.
* Conklin, W. A., Davis, R. L., & Williams, D. (2018). Principles of Computer Security: CompTIA Security+ and Beyond.
* Syngress. (2015). Physical Security Principles.

**Medios Electrónicos**

* Lenguaje PHP. Recuperado en [2023, 14 de junio 9:30 a.m.] de https://es.wikipedia.org/wiki/PHP
* Lenguaje Dart. Recuperado en [2023, 14 de junio 10:30 a.m.] de https://es.wikipedia.org/wiki/Dart
* Gestor de base de datos PostgreSQL. Recuperado en [2023, 14 de junio 5:54 p.m.] de https://es.wikipedia.org/wiki/PostgreSQL
* Gestión Documental, Recuperado en [2023, 14 de junio 9:22 p.m.] de http://onegolive.com/es/faq/gestion-documental/ventajas-gestion-documental
* Norma ISO9001. Recuperado en [2023, 14 de junio 9:54 p.m. de http://www.normas9000.com/condiciones-iso-9001.html