



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO SUPERIOR DE LIBRES
Organismo Público Descentralizado del Gobierno del Estado de Puebla

“SISTEMA DIGITAL DE GESTIÓN DE HISTORIALES MÉDICOS OBSTÉTRICOS”

OPCIÓN I

TESIS PROFESIONAL

**QUE PARA OBTENER EL TÍTULO DE:
LICENCIATURA EN INGENIERÍA EN SISTEMAS COMPUTACIONALES**

**PRESENTA:
LORENA PIOQUINTO HERNÁNDEZ**

LIBRES, PUEBLA, ABRIL, 2025.



Educación
Secretaría de Educación Pública



TECNOLOGÍA
SUPERIOR DE LIBRES



Instituto Tecnológico Superior de Libres

Libres, Puebla, **19960192**

Asunto: Liberación de proyecto para la titulación integral

MTRA. BERENICE VICTORIA CORTE
ENCARGADA DE LA JEFATURA DEL DPTO.
DE ESTUDIOS PROFESIONALES
PRESENTE

Por este medio informo que ha sido liberado el siguiente proyecto para la titulación integral:

Nombre del estudiante y/o egresado: Lorena Pioquinto Hernández
Carrera: Licenciatura en Ingeniería en Sistemas Computacionales
No. de control: 19960192
Nombre del proyecto: " Sistema Digital de Gestión de Historiales Médicos Obstétricos"
Producto: I, TESIS PROFESIONAL

Agradezco de antemano su valioso apoyo en esta importante actividad para la formación profesional de nuestros egresados.

ATENTAMENTE

"Por una cultura científica, tecnológica y sustentable"



PA

ING. IVETTE LOBATO GONZÁLEZ
ENCARGADA DE LA DIVISIÓN DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

S.E.P. INSTITUTO TECNOLÓGICO

SUBDIRECCIÓN
ACADÉMICA

M.C. Sócrates Romero Reyes
Asesor

Ing. Manuel Hernández Herrera
Revisor

Dra. Mariana Lobato Báez
Revisor

PA
Ing. Ivette Lobato González
Revisor

ccp. Archivo



2025
Día de
La Mujer
Indígena

Carrón 854 s/n, Barrio de Totela, Libres, Puebla, C.P. 73700
Tel. 276 144 3125 Ext. 00 E-mail: dl@iteslibres.mx
E-mail: dl@iteslibres.mx | tl@iteslibres.mx | tl@iteslibres.mx



INSTITUTO
TECNOLÓGICO
SUPERIOR DE LIBRES



CONSEJO DE
PLANEACIÓN



ÍNDICE GENERAL

ÍNDICE GENERAL	iii
ÍNDICE DE TABLAS.....	vi
ÍNDICE DE FIGURAS	vii
RESUMEN	ix
INTRODUCCIÓN.....	x
CAPÍTULO I GENERALIDADES	11
1.1 Introducción	12
1.2 Planteamiento del problema.....	13
1.3 Objetivos	14
1.3.1 Objetivo General	14
1.3.2 Objetivos Específicos	14
1.4 Hipótesis.....	15
1.5 Justificación	16
1.6 Alcances y limitaciones	17
1.6.1 Alcances.....	17
1.6.2 Limitaciones	18
CAPÍTULO II MARCO TEÓRICO	19
2.1 Introducción	20
2.2 ¿Qué es un sistema de gestión de historiales médicos?.....	20
2.2.1 Historial Médico.....	20
2.2.2 Digitalización en la Atención Médica	20
2.3 Sistemas de Gestión de Información Médica	20
2.4 Electrocardiograma	21
2.5 Obstetricia	21

2.6 Patología	21
2.7 Internet	22
2.8 Tecnología.....	22
2.9 Back End	22
2.9.1 Python	22
2.9.2 Java.....	22
2.10 Frontend	23
2.10.1 JavaScript.....	23
2.10.2 React.....	23
2.11 ¿Qué es CSS?	23
2.12 HTML.....	23
2.13 jQuery.....	24
2.14 Gestor de código	24
2.14.1 Visual Studio Code	24
2.15 Gestor de bases de datos	24
2.15.1 MySQL	25
2.16 Servidor web.....	25
2.16.1 Servidor local.....	25
2.16.2 Dirección IP	25
2.17 CRUD	26
CAPÍTULO III METODOLOGÍA	27
3.1 Introducción	28
3.2 Metodología para el desarrollo de la página web para análisis de imágenes de electrocardiogramas y ecocardiogramas (ECG).....	28
3.3 Fase 1: Definición de Funciones Esenciales	29
3.3.1 Etapa 1. Definición de roles del equipo	29
3.3.2 Etapa 2: Diseñar interfaz	33
3.3.3 Etapa 3: Crear la base de datos.....	39

3.4 Fase 2: Desarrollo de la página web	41
3.4.1 Etapa 1: Herramientas de desarrollo	41
3.4.2 Etapa 2: Gestión de base de datos	45
3.5 Fase 3: Integración de Inteligencia Artificial	48
3.5.1 Etapa 1: Procesamiento de señales de ECG.	48
3.5.2 Pre procesamiento de datos.....	49
3.5.3 Diseño de la red neuronal	49
3.5.4 Etapa 2: Interpretación de datos	52
3.6 Fase 4: Implementación y Alojamiento	53
CAPÍTULO IV ANÁLISIS Y DISCUSIÓN DE RESULTADOS.....	57
4.1 Introducción	58
4.2 Implantación y creación de la página web	58
4.3 Pruebas Técnicas.....	62
4.3.1 Validación de datos	62
4.3.2 Pruebas de Configuración	64
4.3.3 Análisis Experimental	64
4.4 Red Neuronal (CNN)	64
4.4.1 Pruebas realizadas en una red neuronal para el procesamiento de imágenes de electrocardiogramas de madres embarazadas	64
4.5 Procesamiento de Imágenes y Espectrogramas para Electrocardiogramas Fetales	70
4.6 Análisis de Riesgos	73
Conclusiones y recomendaciones	75
Referencias	78
Glosario de Términos.	81
Glosario de Acrónimos	82

ÍNDICE DE TABLAS

Tabla 3.1 Plan de sprint.....	31
Tabla 3.2 Product backlog	32
Tabla 3.3 Sprint backlog.....	33
Tabla 3.4 Plan de sprint.....	37
Tabla 3.5 Product backlog	39
Tabla 3.6 Sprint back log	39
Tabla 3.7 Actividades Scrum Team.....	41
Tabla 3.8 Product Backlog diseño	44
Tabla 3.9 Sprint backlog.....	45
Tabla 3.10 Actividades Scrum Team.....	46
Tabla 3.11 Product Backlog	47
Tabla 3.12 Sprint backlog.....	48
Tabla 3.13 Actividades Scrum Team.....	49
Tabla 3.14 Product Backlog	51
Tabla 3.15 Sprint backlog.....	51
Tabla 3.16 Plan de sprint de actividades.....	55
Tabla 3.17 Sprint backlog.....	56

ÍNDICE DE FIGURAS

Figura 3.1 Metodología para la creación de página web para historiales médicos	28
Figura 3.2 Fases de la metodología	29
Figura 3.3 Estructura orgánica de roles.....	30
Figura 3.4 Login de la página	35
Figura 3.5 Captura de datos para generar historial clínico	36
Figura 3.6 Búsqueda de historial médico	36
Figura 3.7 Historial de pacientes para cargar en PDF.....	37
Figura 3.8 Descarga de gestor para la base de datos de historiales.....	40
Figura 3.9 Base de datos de historiales médicos	40
Figura 4.1 Registro en GitHub	58
Figura 4.2 Archivos subidos al hosting	59
Figura 4.3 Inicio de sesión de la página	60
Figura 4.4 Apartado para captura de datos de pacientes.....	60
Figura 4.5 Modulo para búsqueda de pacientes	61
Figura 4.6 Módulo para descargar el historial del paciente	61
Figura 4.7 Carga de archivo CSV en la página	62
Figura 4.8 Prueba en Postman.....	63
Figura 4.9 Prueba de listar historiales guardados	63
Figura 4.10 Prueba archivo CSV subido correctamente.....	64
Figura 4.11 Capas de red neuronal para el procesamiento de las señales.....	65
Figura 4.12 Accuracy de optimizador Adam.....	66
Figura 4.13 Loss optimizador Adam	67
Figura 4.14 Accuracy de optimizador SGD	67
Figura 4.15 Loss de optimizador SGD.....	68

Figura 4.16 Accuracy de optimizador Adamax	68
Figura 4.17 Loss optimizador Adamax	69
Figura 4.18 Accuracy optimizador RMSprop	69
Figura 4.19 Loss de optimizador RMSprop	70
Figura 4.20 Espectrograma 1	71
Figura 4.21 Espectrograma 2	72
Figura 4.22 Espectrograma 3	72
Figura 4.23 Espectrograma 4	73

RESUMEN

El presente proyecto se delimita al desarrollo de un sistema Backend para gestionar la información básica de pacientes embarazadas y sus respectivos electrocardiogramas fetales. El propósito principal es facilitar el acceso y manejo de los datos médicos, proporcionando una interfaz que permita a los profesionales de la salud introducir el nombre o ID del paciente para visualizar sus datos y cargar archivos CSV de electrocardiogramas fetales.

Para lograr este objetivo, se implementó una solución utilizando tecnologías como Python, JavaScript, y el framework React. Se emplearon bases de datos NoSQL, específicamente MongoDB, para almacenar la información, y GitHub para la gestión del código y control de versiones. El proyecto se llevó a cabo bajo la metodología ágil Scrum, con sprints iterativos y retroalimentación para asegurar un desarrollo flexible y ajustado a las necesidades.

Este estudio se justifica debido a la necesidad de sistemas médicos más eficientes que integren y almacenen información de pacientes de manera segura y accesible. Se fundamenta en la aplicación de las tecnologías de la información para la mejora de los servicios médicos, con base en investigaciones sobre desarrollo de software y gestión de datos médicos.

La investigación tiene un diseño basado en el desarrollo experimental de software. La población objeto del estudio se delimita a los profesionales de la salud que manejan información de pacientes materno-fetales.

INTRODUCCIÓN

En la actualidad, la gestión eficiente de datos en el ámbito de la salud es fundamental para garantizar una atención de calidad y optimizar los procesos clínicos. Esta tesis se centra en el desarrollo de un sistema Backend (parte invisible de un sitio web o aplicación que se encarga de la lógica y el procesamiento de datos para que funcione correctamente.) para un formulario sencillo que permite a los profesionales de la salud acceder y gestionar información básica de los pacientes. A través de este sistema, se busca facilitar no solo la visualización, de datos esenciales, sino también el acceso a historiales médicos, así como diversos análisis clínicos que incluirán química sanguínea, presión arterial, oximetría de pulso, electrocardiogramas y ecocardiogramas.

Para lograr este objetivo, se emplearán tecnologías avanzadas como Python y Java para el desarrollo de Backend, junto con JavaScript y el framework React (conjunto de conceptos, prácticas y criterios que sirven como referencia para desarrollar un proyecto) para el frontend (aquello que ven los usuarios e incluye elementos visuales, como botones, casillas de verificación, gráficos y mensajes de texto). La elección de MongoDB (sistema de gestión de bases de datos (DBMS) no relacional de código abierto y gratuito) y NoSQL (se refiere a tipos de bases de datos no relacionales que almacenan datos en un formato distinto a las tablas relacionales diseñado para administrar conjuntos de datos grandes y complejos) como base de datos responde a la necesidad de manejar grandes volúmenes de información de manera flexible y escalable, permitiendo una rápida recuperación y almacenamiento eficiente de datos.

Además, se implementará la metodología Scrum, lo que permitirá una colaboración continua entre los miembros del equipo y una adaptación constante a los cambios que pueden surgir durante el desarrollo del proyecto. Este enfoque no solo optimiza el tiempo de entrega, sino que también asegura que las necesidades del usuario final sean atendidas de manera efectiva.

A través de este trabajo, se espera contribuir al campo de la salud, con una solución digital, ofreciendo una herramienta que no solo mejorara la gestión de información médica, también facilitara la toma de decisiones clínicas.

CAPÍTULO I

GENERALIDADES

1.1 Introducción

En el ámbito de la salud, la gestión eficiente de la información médica es crucial para garantizar una atención de calidad y mejorar los procesos clínicos. En este contexto, el presente proyecto se centra en el desarrollo de un sistema Backend orientado a la administración de datos de pacientes embarazadas y sus electrocardiogramas fetales. Su propósito principal es proporcionar una plataforma que facilite el acceso y manejo de historiales médicos, permitiendo a los profesionales de la salud visualizar información esencial e integrar análisis clínicos de manera estructurada.

Para lograr este objetivo, se ha desarrollado una solución tecnológica basada en lenguajes de programación como Python y JavaScript, junto con el framework React para la interfaz de usuario. Como base de datos, se ha implementado MongoDB, una solución NoSQL que permite un almacenamiento flexible y escalable de grandes volúmenes de información médica. Además, el control de versiones se gestiona mediante GitHub, asegurando un desarrollo colaborativo y organizado.

La metodología ha sido clave en la planificación del proyecto, permitiendo iteraciones constantes y ajustes en función de las necesidades del usuario final. Esta metodología garantiza una entrega eficiente y adaptada a las exigencias del sector médico.

Con esta iniciativa, se busca contribuir al ámbito de la salud mediante una herramienta tecnológica que optimice la gestión de datos clínicos y facilite la toma de decisiones médicas. La integración de tecnologías avanzadas y el enfoque en la accesibilidad y seguridad de la información representan un paso significativo en la mejora de los servicios médicos.

1.2 Planteamiento del problema

En la actualidad, las tecnologías de la información y las comunicaciones (TIC's) ha jugado un papel fundamental en la mejora de la atención médica, permitiendo el acceso rápido y eficiente a los datos de los pacientes. Sin embargo, muchas instituciones de salud aún carecen de sistemas adecuados que permitan gestionar y visualizar de forma eficiente información clave como los datos básicos del paciente y resultados de exámenes clínicos en tiempo real. Un área de especial interés es el monitoreo fetal, donde las señales fisiológicas como los electrocardiogramas (ECG) fetales, Ecocardiogramas, Presión Arterial, oximetría, entre otros son una herramienta crucial para el seguimiento de la salud del feto. Sin embargo, el manejo de estos datos aún presenta desafíos significativos en términos de accesibilidad, integración y almacenamiento.

El problema a resolver es la falta de un sistema que permita gestionar la información de los pacientes, incluyendo la capacidad de cargar y visualizar archivos de señales fisiológicas fetales, de manera eficiente y centralizada. Además, se necesita un Backend robusto y flexible que soporte la carga y visualización de esta información, garantizando la integridad de los datos y su fácil acceso por parte de los médicos y profesionales de la salud.

Este proyecto plantea el desarrollo de un sistema Backend que es usado en un formulario que permite a los médicos introducir un identificador o nombre del paciente, y acceder de inmediato a los datos básicos del mismo, así como cargar y visualizar archivos que contengan señales fisiológicas como por ejemplo electrocardiogramas fetales, ecocardiogramas, presión arterial, oxigenación, entre otros. Se realizó el uso de tecnologías como Python (lenguaje de programación de alto nivel, interpretado y orientado a objetos, que se utiliza para crear aplicaciones web, software, ciencia de datos y machine learning), JavaScript (lenguaje de secuencias de comandos que te permite crear contenido de actualización dinámica), y frameworks de desarrollo web como React para la interfaz de usuario. Además, se utilizarán bases de datos NoSQL como MongoDB para el almacenamiento de los datos del paciente, y herramientas de control de versiones como Git y GitHub (plataforma en la nube que permite almacenar, compartir y trabajar en código junto con otros usuarios) para gestionar el desarrollo colaborativo. La metodología se adaptó para organizar el proyecto, asegurando un desarrollo ágil e iterativo.

El desarrollo de este sistema resolvió la falta de herramientas adecuadas para la gestión de información crítica en el ámbito de la salud materno-fetal, contribuyendo a mejorar la calidad del seguimiento médico y la toma de decisiones oportunas.

1.3 Objetivos

1.3.1 Objetivo General

Desarrollar un sistema Backend eficiente que permita la gestión de un formulario médico, capaz de visualizar datos básicos del paciente, acceder a sus archivos clínicos y señales fisiológicas del feto, empleando tecnologías como Python, Java, JavaScript, React, MongoDB, NoSQL, y herramientas de control de versiones como Git y GitHub. Además, se incorporará la funcionalidad de cargar archivos de electrocardiogramas en formato CSV.

1.3.2 Objetivos Específicos

- Investigar los diferentes análisis clínicos referentes a mujeres embarazadas.
- Analizar e investigación de las tecnologías compatibles para realizar una página web que integre el historial clínico de mujeres embarazadas
- Implementar un formulario en el frontend que permita la visualización de los datos básicos del paciente, utilizando React para la interfaz de usuario.
- Integrar una funcionalidad en el sistema que permita cargar y visualizar archivos de electrocardiogramas en formato CSV, vinculados a cada paciente de manera automática, utilizando bases de datos NoSQL como MongoDB.
- Desarrollar un componente para la visualización de imágenes del electrocardiograma del feto.
- Garantizar la integridad y seguridad de los datos almacenados en la base de datos, siguiendo buenas prácticas de desarrollo backend y configuraciones adecuadas en NoSQL para la consulta eficiente de la información.

1.4 Hipótesis

La implementación de un sistema Backend robusto y escalable, en la creación de un gestor web para historial médico para embarazadas que ayudara a eficientar el manejo de información médico, utilizando tecnologías de desarrollo modernas como Python, Java, JavaScript, el framework React, bases de datos NoSQL como MongoDB, y herramientas de control de versiones como Git y GitHub, permitirá la creación de un formulario clínico interactivo que optimice la gestión de la información médica básica de los pacientes, incluyendo la visualización de archivos y electrocardiogramas fetales. Se espera que este sistema proporcione una mejora significativa en la eficiencia, precisión y seguridad en la gestión de datos médicos, con un impacto directo en el seguimiento y monitoreo de la salud fetal a través del electrocardiograma fetal (ECG), una herramienta crucial para la evaluación de la salud del feto.

Los electrocardiogramas fetales (ECG) son fundamentales para el monitoreo de la salud del feto durante el embarazo, ya que permiten detectar posibles irregularidades en el ritmo cardíaco fetal, como arritmias o taquicardias, que pueden ser indicativos de problemas cardiovasculares o complicaciones que requieran intervención médica. Esta tecnología ofrece una manera no invasiva de controlar el bienestar del feto, proporcionando información valiosa sobre su estado de oxigenación y el funcionamiento de su corazón, factores cruciales para determinar si se necesita algún tipo de tratamiento prenatal o intervención durante el parto.

El uso de una base de datos NoSQL como MongoDB permitirá un almacenamiento y recuperación más rápida de los datos de los pacientes y sus electrocardiogramas, en comparación con una base de datos relacional tradicional.

El uso de inteligencia artificial en el diagnóstico de enfermedades crónicas no tendrá un impacto significativo en la precisión de los resultados.

1.5 Justificación

El presente proyecto tiene como objetivo el poder desarrollar un sistema de Backend para un formulario sencillo que permita gestionar información básica de pacientes, así como facilitar el acceso al archivo médico, incluyendo imágenes de electrocardiogramas fetales. En el contexto donde la gestión eficiente de datos médicos es esencial, este sistema se convierte en una herramienta vital para mejorar la atención al paciente y optimizar los procesos clínicos.

La necesidad de implementar este sistema radica en la creciente demanda de soluciones tecnológicas que permitan a los profesionales de la salud acceder a información relevante de manera rápida y segura. Actualmente, muchos registros son gestionados manualmente, lo que puede llevar a errores y demoras en la atención. Al utilizar tecnologías como Python, Java, JavaScript y frameworks de desarrollo web como React, se garantiza una interfaz amigable y funcional. Las bases de datos NoSQL como MongoDB permitirán almacenar y recuperar datos de forma ágil y eficiente.

Además, con el uso de herramientas de control de versiones como Git y GitHub, se facilitará la colaboración entre los desarrolladores del equipo asegurando un flujo de trabajo organizado.

Por consiguiente, la capacidad para subir archivos en formato CSV del electrocardiograma no solo simplificará la carga de datos médicos, sino que también mejorará la precisión en el análisis y seguimiento del estado del feto. Este sistema no solo responde a una necesidad crítica en el ámbito médico, sino que también representa un avance significativo hacia la digitalización y modernización de los servicios de salud.

El desarrollo de un sistema Backend para gestionar la información básica de pacientes y visualizar electrocardiogramas fetales se justifica por la creciente necesidad de mejorar la eficiencia en la atención médica, especialmente en el ámbito de la salud materno-fetal. Muchas instituciones de salud carecen de herramientas tecnológicas que permitan una gestión rápida, segura y eficiente de los datos clínicos, lo que puede derivar en errores médicos, demoras en la toma de decisiones y dificultades para acceder a información crítica en tiempo real.

Este proyecto busca mejorar la gestión de historiales médicos obstétricos mediante la implementación de un sistema que agilice el acceso a la información del paciente. El personal médico podrá acceder de manera rápida y eficiente a los datos básicos del paciente, su historial médico y los resultados de pruebas diagnósticas como el electrocardiograma fetal. Esto optimizará el tiempo de atención, mejorará la calidad del diagnóstico y permitirá una organización más efectiva de los datos clínicos.

La implementación de este sistema digital permitirá:

- Acceso ágil a la información: el sistema permitirá a los profesionales de la salud acceder a los datos de los pacientes de manera rápida y eficiente, lo que optimizará los tiempos de atención y la precisión en los diagnósticos.
- Mejor organización de datos: el almacenamiento digital garantizará una mejor conservación y organización de los historiales médicos, evitando el deterioro o pérdida de documentos físicos.
- Promoción de atención prenatal de calidad: facilitará el seguimiento continuo del embarazo, permitiendo identificar riesgos y adoptar medidas preventivas adecuadas, lo que contribuirá a una atención prenatal más eficiente y de mayor calidad.

Este sistema no solo optimizará los procesos de gestión de la información médica, sino que también contribuirá a mejorar la salud materna e infantil al proporcionar una herramienta eficiente para el seguimiento y diagnóstico en el área de obstetricia.

1.6 Alcances y limitaciones

1.6.1 Alcances

- Mayor precisión en los diagnósticos: el sistema podría permitir una detección más temprana de anomalías fetales, lo que podría llevar a intervenciones médicas más oportunas y mejores resultados para la madre y el feto.
- Toma de decisiones más informadas: los médicos podrían tomar decisiones clínicas más basadas en evidencia al tener acceso a información detallada y visualizada de los electrocardiogramas fetales.

- Mayor eficiencia en el flujo de trabajo: el sistema podría agilizar los procesos de diagnóstico y seguimiento, reduciendo el tiempo que los médicos dedican a tareas administrativas.
- Mejor comunicación entre los profesionales de la salud: el sistema podría facilitar la comunicación y colaboración entre diferentes especialistas involucrados en el cuidado de la embarazada.

1.6.2 Limitaciones

- Dependencia de la calidad de los datos: la precisión del sistema depende en gran medida de la calidad de los electrocardiogramas obtenidos. Factores como el ruido en la señal, la posición fetal y la calidad del equipo pueden afectar la interpretación de los datos.
- Interpretación humana: aunque el sistema proporciona herramientas de visualización y análisis, la interpretación final de los electrocardiogramas sigue siendo responsabilidad del médico.
- Costo: la implementación y mantenimiento de un sistema de este tipo puede implicar costos significativos, tanto en términos de hardware, software como de capacitación del personal.
- Aceptación por parte de los usuarios: la adopción del sistema por parte de los profesionales de la salud puede requerir un proceso de adaptación y capacitación. Es importante considerar factores como la resistencia al cambio y la facilidad de uso de la interfaz.
- Limitaciones tecnológicas: la tecnología actual puede tener limitaciones en cuanto a la resolución de las imágenes, la capacidad de procesamiento de datos y la detección de ciertos tipos de anomalías.

CAPÍTULO II

MARCO TEÓRICO

2.1 Introducción

En el presente capítulo se abordan los conceptos fundamentales del marco teórico referente al proyecto que tiene por nombre “Sistemas digital de gestión de historiales médicos obstétricos”

2.2 ¿Qué es un sistema de gestión de historiales médicos?

Brinda una propuesta para el desarrollo de un sistema en la cual se propone sistematizar el proceso de la redacción y generación de historiales médicos con las finalidades tales como la reducción del tiempo de la generación de historiales y la reducción de recursos físicos, especialmente en lo que respecta al uso del papel, la cual los especialistas de la salud redactan en el transcurso de una consulta médica, esto además implicaría en la optimización en la gestión de los historiales médicos (Ramírez, 2021).

2.2.1 Historial Médico

La historia clínica es una de las formas de registro del acto médico, cuyas cuatro características principales se encuentran involucradas en su elaboración y son: profesionalidad, ejecución típica, objetivo y licitud (Guzmán, 2012).

2.2.2 Digitalización en la Atención Médica

La digitalización de archivos médicos ha transformado el ámbito de la atención sanitaria, prometiendo mejorar la eficiencia, la precisión y el acceso a la información (Otaola Barranquero 2021).

2.3 Sistemas de Gestión de Información Médica

Programas informáticos que ayudan a los profesionales de la salud a realizar tareas relacionadas con la atención al paciente.

- Sistema de Información Clínica (S.I.C.)

Se utiliza para ayudar a los médicos en todos los niveles de la atención médica. Es el sistema principal para la enfermería y médicos en el entorno de los cuidados intensivos.

- Sistema de Información Hospitalaria (HIS)

Es una plataforma que permite el registro de información de pacientes por un equipo multidisciplinario. También puede rastrear y registrar todas las transacciones financieras en una organización de atención médica.

- Sistemas para la administración de la atención médica

Son un tipo de software enfocado en recopilar, almacenar, procesar y analizar información de pacientes en una institución de salud (Vidal Ledo, 2014).

2.4 Electrocardiograma

El EC es un gráfico en el que se estudian las variaciones de voltaje en relación con el tiempo. Consiste en registrar en un formato especialmente adaptado (tiras de papel milimetrado esencialmente), la actividad de la corriente eléctrica que se está desarrollando en el corazón durante un tiempo determinado (en un EC normal no suele exceder los 30 segundos) (Azcona, 2009).

2.5 Obstetricia

La obstetricia es aquella parte de la medicina que trata la gestación, el parto y el puerperio, por su parte, la semiología es la ciencia que estudia síntomas y signos de las enfermedades, incorporando como elementos esenciales la anamnesis y el examen físico, en sus aspectos fundamentales, la semiología en ginecología y obstetricia es similar a lo aprendido en semiología general (D. G, 2019).

2.6 Patología

Es el estudio de la enfermedad, una disciplina que conecta la práctica clínica y la ciencia básica, e implica la investigación de las causas de la enfermedad, así como de los mecanismos subyacentes que dan lugar a los signos y síntomas de presentación del paciente (Kumar, Cotran y Robbins 2008).

2.7 Internet

El término Internet se utiliza para hacer referencia a una red de computadoras interconectadas entre sí a nivel mundial con el objetivo de hacer común información de acceso público (Pedroso Herrera, 2006).

2.8 Tecnología

Conjunto de conocimientos, técnicas, habilidades, métodos y procesos que se utilizan para alcanzar objetivos o solucionar problemas (Ezeiza Pohl, C. E., Rampello, S. M., & Mecca, D. E. 2024).

2.9 Back End

La programación y el acceso a los datos que no son directamente accesibles para el usuario final (Ibarra, 2021).

2.9.1 Python

Lenguaje de programación de alto nivel que se utiliza para crear una amplia variedad de aplicaciones, desde sitios web y juegos hasta análisis de datos y aprendizaje automático (Jindal, 2021).

2.9.2 Java

Lenguaje de programación orientado a objetos y una plataforma informática muy popular y ampliamente utilizada en el mundo digital. Fue creado por Sun Microsystems (ahora propiedad de Oracle) a principios de los años 90 y desde entonces se ha convertido en una de las tecnologías más confiables y robustas para desarrollar aplicaciones (Montavez, 2024).

2.10 Frontend

Todo lo que un usuario ve e interactúa directamente en un sitio web o aplicación. Es la interfaz gráfica de usuario (GUI) que te permite navegar, hacer clic en botones, llenar formularios y ver el contenido de una página (Caballero, JJV,2016).

2.10.1 JavaScript

Es un lenguaje de programación que se utiliza principalmente para crear páginas web interactivas. Es decir, gracias a JavaScript, las páginas web pueden hacer mucho más que simplemente mostrar información estática (Pérez, JE,2019).

2.10.2 React

Es una biblioteca de JavaScript de código abierto, creada y mantenida por Facebook, diseñada específicamente para construir interfaces de usuario de aplicaciones web. Es una herramienta muy popular y ampliamente utilizada en el desarrollo frontend (Gackenhimer, C.,2015).

2.11 ¿Qué es CSS?

CSS, o Hojas de Estilo en Cascada, es un lenguaje que define cómo se presenta un documento HTML o XML en la web. Permite controlar el diseño visual de una página, como colores, fuentes y disposición de elementos, haciendo posible crear diseños responsivos y atractivos. CSS es fundamental para separar el contenido del diseño, lo que facilita la gestión y flexibilidad de los proyectos web (W3C, s.f.).

2.12 HTML

HTML, o Lenguaje de Marcado de Hipertexto, es el estándar utilizado para crear y diseñar páginas web. Define la estructura básica de una página web mediante el uso de etiquetas para elementos como párrafos, encabezados, imágenes y enlaces. Cada elemento HTML describe una parte diferente del contenido, lo que permite a los navegadores mostrarlo correctamente en la pantalla del usuario (MDN, s.f.).

2.13 jQuery

jQuery es una biblioteca de JavaScript rápida, pequeña y rica en funciones que simplifica la manipulación de documentos HTML, el manejo de eventos, la creación de animaciones y la interacción con Ajax. Al abstraer muchas de las complejidades del desarrollo web, jQuery permite a los desarrolladores escribir menos código y lograr más, mejorando la compatibilidad entre navegadores y facilitando la creación de interfaces de usuario interactivas y dinámicas (Parada, 2019).

2.14 Gestor de código

Los sistemas de gestión del código fuente (SCM) son herramientas que facilitan la colaboración y modificación del repositorio de código fuente en proyectos. Funcionan mediante el seguimiento de los cambios realizados en el código a lo largo del tiempo (Unity, s.f).

2.14.1 Visual Studio Code

El IDE de Visual Studio es una plataforma de lanzamiento creativa que se puede utilizar para editar, depurar y compilar código y, finalmente, publicar una aplicación. Además del editor y depurador estándar que ofrecen la mayoría de IDE, Visual Studio incluye compiladores, herramientas de completado de código, diseñadores gráficos y muchas más funciones para mejorar el proceso de desarrollo de software (Visual Studio, 2024).

2.15 Gestor de bases de datos

Un gestor de bases de datos (DBMS, por sus siglas en inglés) es un software que facilita la creación, gestión y manipulación de bases de datos. Permite a los usuarios almacenar, organizar y recuperar datos de manera eficiente. Los DBMS proporcionan herramientas para definir la estructura de los datos, realizar consultas, actualizar la información y mantener la integridad y seguridad de los datos" (Oracle, s.f.).

2.15.1 MySQL

MySQL es un sistema de gestión de bases de datos relacional de código abierto que utiliza el lenguaje de consulta estructurado (SQL) para gestionar los datos. Es conocido por su alto rendimiento, fiabilidad y facilidad de uso, y es ampliamente utilizado en aplicaciones web para almacenar y recuperar datos (MySQL, s.f.).

2.16 Servidor web

Un servidor web es un software o hardware que almacena, procesa y entrega páginas web a los usuarios a través de la red. Funciona mediante el protocolo HTTP (o HTTPS) para recibir solicitudes de los navegadores web y enviarles las respuestas en forma de documentos HTML, imágenes y otros recursos multimedia. Los servidores web gestionan el tráfico de datos y aseguran que el contenido web sea accesible y funcional para los usuarios en línea (Apache Software Foundation, s.f.).

2.16.1 Servidor local

Un servidor local es una instancia de un servidor web que se ejecuta en una máquina local (generalmente el ordenador del desarrollador) para probar y desarrollar aplicaciones web antes de desplegarlas en un servidor de producción. Permite a los desarrolladores verificar y depurar el funcionamiento de sus aplicaciones en un entorno controlado, sin necesidad de estar conectado a internet" (Stack Overflow, s.f.).

2.16.2 Dirección IP

Una dirección IP (Protocolo de Internet) es una etiqueta numérica asignada a cada dispositivo conectado a una red que utiliza el Protocolo de Internet para la comunicación. La dirección IP permite que los dispositivos se identifiquen y localicen entre sí en la red, facilitando el intercambio de datos. Existen dos versiones principales de direcciones IP: IPv4, que utiliza direcciones de 32 bits, e IPv6, que usa direcciones de 128 bits para permitir una mayor cantidad de direcciones únicas (IETF, s.f.).

2.17 CRUD

CRUD es un acrónimo que representa las operaciones básicas que se pueden realizar en una base de datos o sistema de gestión de datos: Crear (Create), Leer (Read), Actualizar (Update) y Eliminar (Delete). Estas operaciones permiten a los usuarios interactuar con los datos almacenados, facilitando la administración y manipulación de la información dentro de una aplicación (IBM, s.f.).

CAPÍTULO III

METODOLOGÍA

3.1 Introducción

El presente capítulo describe la metodología utilizada para el desarrollo de una página web destinada al análisis de imágenes de electrocardiogramas (ECG). A través de un enfoque estructurado en fases, se busca optimizar la gestión de historiales médicos y mejorar la eficiencia en el análisis de registros médicos mediante la integración de tecnologías avanzadas. A continuación, se detallan las fases del proyecto y las tecnologías empleadas en cada una de ellas.

3.2 Metodología para el desarrollo de la página web para análisis de imágenes de electrocardiogramas y ecocardiogramas (ECG)

En la figura 3.1 y 3.2 se describe la metodología para la creación de la página web para historiales médicos, así como sus diversas fases.

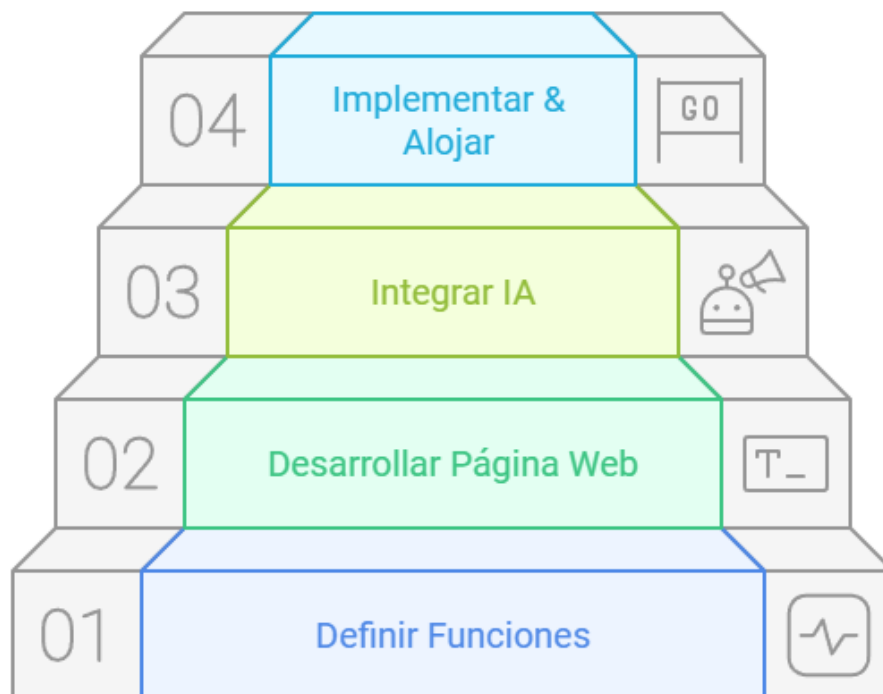


Figura 3.1 Metodología para la creación de página web para historiales médicos

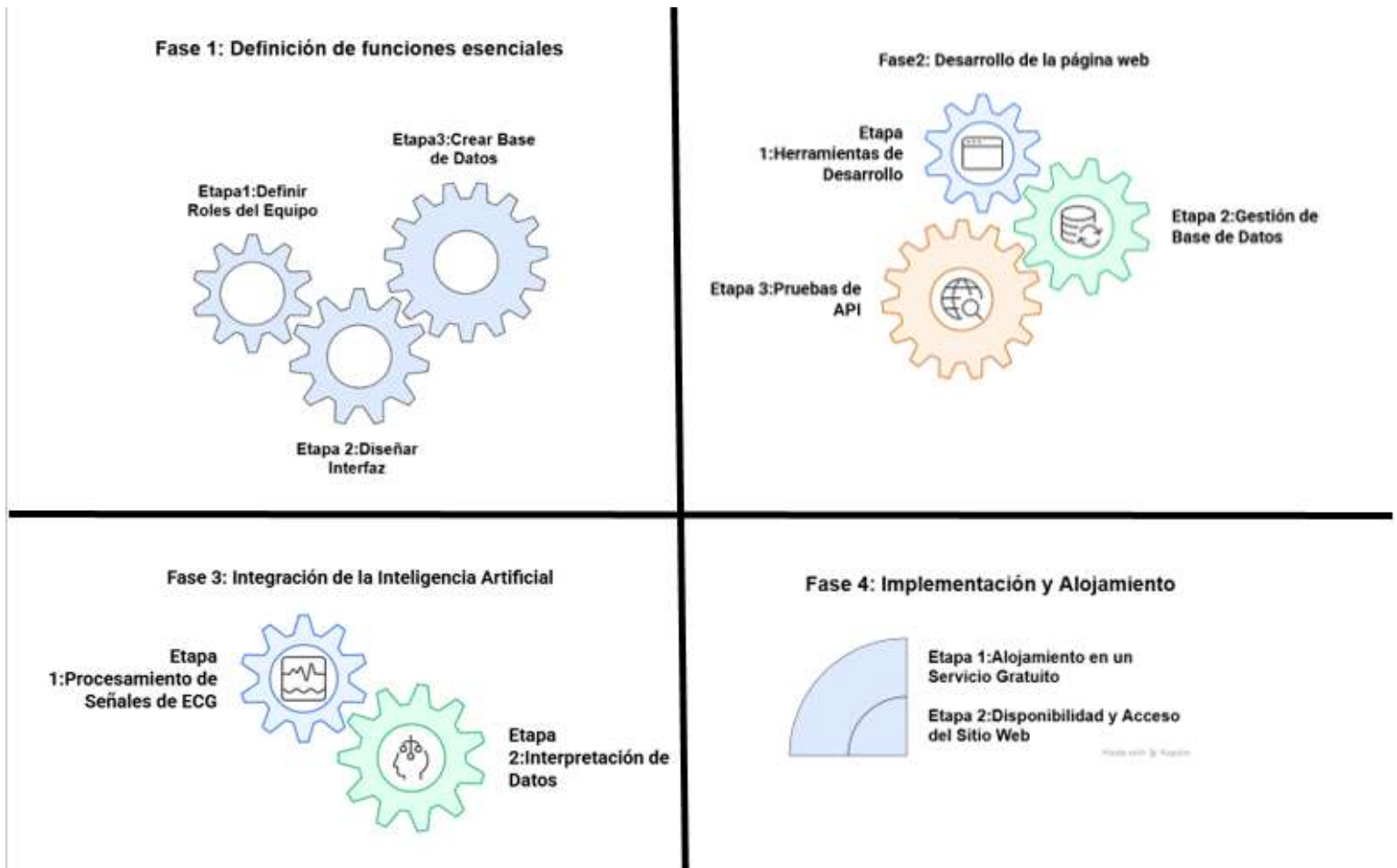


Figura 3.2 Fases de la metodología

3.3 Fase 1: Definición de Funciones Esenciales

- **Roles del equipo:**
 - Product Owner
 - Scrum Master
 - Development Team

3.3.1 Etapa 1. Definición de roles del equipo

Una de las características que conforman a la Metodología son los roles, para dar inicio con el desarrollo del proyecto es necesario definir a cada uno, recordando que estos roles están divididos por: el Product Owner, el Scrum Master y el Development Team. Es importante llevar a cabo este proceso de identificación y asignación de roles, ya que sin

ello no es posible llevar a cabo esta metodología en la figura 3.3 se puede observar la estructura orgánica del equipo de trabajo.



Figura 3.3 Estructura orgánica de roles

En el proyecto de la página web para la creación de historiales médicos para mujeres embarazadas al Development Team se le asignó la responsabilidad de lograr mantener la organización de la página mediante módulos de manera eficiente.

- **Funciones esenciales de la página web:**
 - Diseño de la interfaz web.
 - Creación de una base de datos para almacenar información de pacientes.
- **Objetivo:** Optimizar el tiempo de búsqueda mediante un método práctico.

Sprint planning reunión

Durante el desarrollo de este sprint, se llevó a cabo la fase de análisis a partir de la primera solicitud de requerimientos de la aplicación.

Objetivos: Comprender los requisitos pedidos para la realización del diseño de la página web y los módulos

Descripción: En la reunión de planificación del sprint, se dedicará a examinar detalladamente los requisitos del proyecto, con el propósito de determinar las tareas que deben ejecutarse en el próximo sprint.

Desglose de actividades

En la tabla 3.1 se muestra el desglose de actividades del primer sprint

Persona asignada	Descripción	Tarea a realizar	Tiempo
Lorena Pioquinto Hernández	Realizar el diseño de los módulos de la página web	Analizar las necesidades del usuario	5 horas
	Propuestas de diseño de los módulos de la pagina	Revisión de las propuestas y creación del diseño	5 horas
		Evaluación del modelo	4 horas
	Revisión de herramientas a utilizar	Analizar el diseño para su correcta implementación	4 horas

Tabla 3.1 Plan de sprint

Historias de usuario

Las historias de usuario se implementaron conforme a los requerimientos de la primera fase del proyecto para la implementación del desarrollo de la página web y la revisión de la misma para detectar errores y corrección.

Descripción:

- Identificar los requisitos del sitio web funcionales y no funcionales.
- Scrum Master: aquí se aprueban y revisan los requisitos del proyecto, y revisar cada una de las tareas asignadas.

- Development Team: se recopiló e investigó los requisitos funcionales del proyecto, además de determinar las herramientas a emplear en el desarrollo.
- Se analizó el diseño de la página web y la aplicación.

Product backlog

Se construye a partir de historias de usuario, las cuales han sido desarrolladas y clasificadas en categorías para presentar el backlog de manera más accesible. Esto facilita la comprensión de cómo se llevará a cabo esta fase. Se muestra en la tabla 3.2, las horas que se emplearon para cumplir con el análisis de requerimientos, los requerimientos funcionales y las mejoras que se desarrollaron en la página web de acuerdo a lo establecido con cada revisión.

Descripción	Prioridad	Complejidad	Número de Sprint	Duración
Análisis de los requerimientos de usuario involucrados el product Owner, el development	5	4	1	4 horas
Requerimientos funcionales	5	5	1	3 horas
Requerimientos no funcionales	5	5	1	3 horas
Mejoras de la página web	5	5	1	5 horas
Funcionamiento según el desarrollo de actividades	5	5	1	4 horas

Tabla 3.2 Product backlog

Sprint Backlog

En la tabla 3.3 se observa que la responsabilidad del equipo de desarrollo es promover una comunicación continua, asegurando que cualquier cambio o ajuste necesario en la interfaz o en la base de datos se realice de manera efectiva. Esto permite una colaboración fluida entre los miembros del equipo, garantizando que el sprint avance de acuerdo a los plazos y expectativas del proyecto como lo detalla

Producto	Pendiente	En progreso	Finalizado
Sprint 1	Análisis de aplicaciones	Historias de usuario	Requerimientos funcionales
Objetivo de sprint: Analizar y comprender los requisitos del proyecto para determinar las tareas que se deben completar durante el siguiente sprint.		Análisis de requerimientos no funcionales	

Tabla 3.3 Sprint backlog

3.3.2 Etapa 2: Diseñar interfaz

Sprint planning reunión

Durante esta reunión se llevó acabo el diseño de la interfaz para los diferentes módulos, analizando los requerimientos del Product Owner, para la fácil navegación y captura de datos que generaran los historiales médicos.

Objetivos: configurar la interfaz de la página web para la gestión de historiales clínicos de mujeres embarazadas de manera que satisfaga los requisitos y necesidades identificados. El diseño de la interfaz debe ofrecer una experiencia de usuario intuitiva, atractiva y eficiente, facilitando el acceso a las funciones clave, como la carga de archivos CSV, la visualización de ECG y el acceso a los historiales médicos de los pacientes.

Descripción: en la reunión de planificación del sprint, la persona encargada del desarrollo se enfocó en el análisis y diseño detallado de la página web, considerando aspectos fundamentales como la paleta de colores, la disposición de los elementos y la usabilidad de la interfaz. Se discutió cómo hacer que la plataforma sea fácil de usar para los especialistas médicos, asegurando que el acceso a los historiales clínicos y la visualización de los ECG sea claro y eficiente.

Durante esta sesión, se realizó una discusión sobre los requisitos específicos de la página, abordando las funcionalidades que deben ser implementadas y las prioridades de desarrollo. Esto incluyó un análisis de las dependencias entre las diversas secciones del sitio web, como la carga de archivos y el manejo de la base de datos, con el fin de garantizar que todas las partes del sistema funcionen de manera coherente.

Este proceso permitió al encargado del desarrollo obtener una visión clara de los desafíos técnicos y las oportunidades de mejora en el proyecto, asegurando que se tomen decisiones informadas y se eviten posibles obstáculos en el futuro.

Este enfoque no solo fomentó la coordinación efectiva entre los miembros del equipo, sino que también garantizó que cada tarea realizada contribuyera de manera significativa a la creación de una herramienta útil y eficiente para los profesionales de la salud, optimizando el seguimiento de los historiales médicos y ECG de las pacientes.

La interfaz debe facilitar una gestión eficiente y segura de la información clínica, alineándose estrechamente con las expectativas de los usuarios finales. Esto implicó un diseño centrado en el usuario que optimice el acceso rápido a los datos relevantes, promueva la precisión en los registros médicos y mejore la experiencia general de uso.

A continuación, se muestran imágenes de la maquetación de la página web para generar historiales médicos.

En la figura 3.4 se aprecia la maquetación del inicio de la página, que es el login para acceder a la captura de y que es la interfaz principal.

[INICIO](#) [BUSQUEDA](#) [HISTORIAL PACIENTE](#) [LOGIN](#)

INICIAR SESIÓN / REGISTRO

USUARIO

CONTRASEÑA

ROL

INGRESAR

REGISTRAR

Figura 3.4 Login de la página

Se muestra en la figura 3.5 la maquetación de lo que es el inicio de la captura de datos del paciente, para generar su historial médico.

INICIO
BUSQUEDA
HISTORIAL PACIENTE
LOGIN

NOMBRE DEL PACIENTE

FECHA

NUMERO DE SEGURO SOCIAL

ID DEL PACIENTE

EDAD

TELEFONO

SEXO

ARCHIVO CSV

ELEGIR ARCHIVO
NO SE GUARDO NINGUN ARCHIVO

Figura 3.5 Captura de datos para generar historial clínico

La figura 3.6 detalla el modulo para la búsqueda de los historiales médicos.

INICIO
BUSQUEDA
HISTORIAL PACIENTE
LOGIN

BUSQUEDA DE HISTORIAL MÉDICO

Figura 3.6 Búsqueda de historial médico

En la figura 3.7 se aprecia el módulo en el que se pueden descargar los historiales de pacientes.



Figura 3.7 Historial de pacientes para cargar en PDF

Desglose de actividades

En la tabla 3.4 se detallan los requisitos funcionales y no funcionales del diseño de la página web y las propuestas realizadas para mejorar el diseño de interfaz.

Asignación	Descripción	Tareas	Horas aproximadas
Lorena Pioquinto Hernández	Gestionar datos de presentación	Analizar los requerimientos no funcionales de la aplicación	5
	Diseño de propuestas de interfaz	Revisión de interfaz	5
	Gestionar requerimientos funcionales de la aplicación	Analizar las necesidades del usuario	5
	Analizar las herramientas a utilizar	Análisis del diseño	5

Tabla 3.4 Plan de sprint

Historias de usuario

- Identificación de los requisitos de la página web funcionales como no funcionales.

- Scrum Master: En esta reunión, se encargó de aprobar y revisar los requisitos del proyecto.
- Development Team: Se encargó de recopilar e investigar los requisitos funcionales del proyecto, además de determinar las herramientas a emplear en el desarrollo.
- Análisis de colores y diseños para tener una visión clara de la ejecución, así como de la maquetación de la página web y la aplicación.

Product backlog

Es necesario considerar que el product backlog se construye a partir de historias de usuario, las cuales han sido desarrolladas y clasificadas en categorías para presentar el backlog de manera más accesible. Esto facilita la comprensión de cómo se llevará a cabo esta fase. En la tabla 3.5 se pueden observar las horas que se llevaron a cabo para las diferentes tareas de cada integrante del equipo.

Descripción	Prioridad	Complejidad	No. Sprint	Duración
Análisis de los requerimientos de usuario con ayuda del product Owner, el development	5	4	1	4 horas
Requerimientos funcionales de la página	5	4	1	4 horas
Requerimientos no funciones del sitio web	5	4	1	3 horas
Desarrollo de mejoras	5	4	1	4 horas
Implementación de	5	4	1	4 horas

funcionamiento según el				
desarrollo de actividades				

Tabla 3.5 Product backlog

Sprint backlog

Las visualizaciones integrales de todas las tareas comprendidas son de importancia en el sprint actual, y su gestión recae en la responsabilidad de la persona de desarrollo promoviendo una comunicación eficiente y una colaboración efectiva por parte de la persona de desarrollo como lo detalla la tabla 3.6.

Producto	Pendiente	En progreso	Finalizado
Sprint 1	Análisis y actualizar página	Historias de usuario	Requerimientos funcionales
Objetivo de sprint: Analizar y comprender los requisitos del proyecto para determinar las tareas que se deben completar durante el siguiente sprint.		Análisis de requerimientos no funcionales	

Tabla 3.6 Sprint back log

3.3.3 Etapa 3: Crear la base de datos

En esta etapa se creó la base de datos en MongoDB para registrar los datos capturados de los pacientes relacionados con su historial médico. Este proceso permite contar con un sistema organizado donde se pueda almacenar, visualizar y gestionar la información de cada paciente de manera eficiente. Para ello, se seleccionó un gestor adecuado que facilite la creación y administración de la base de datos, asegurando un control preciso sobre el registro individual de cada paciente. Para este proceso, se descargó el gestor

elegido y se creó la base de datos llamada Historial pacientes en una carpeta en específico para nuestros pacientes.

La figura 3.8 muestra la carpeta en la cual se guardan los pacientes registrados en el gestor MongoDB.

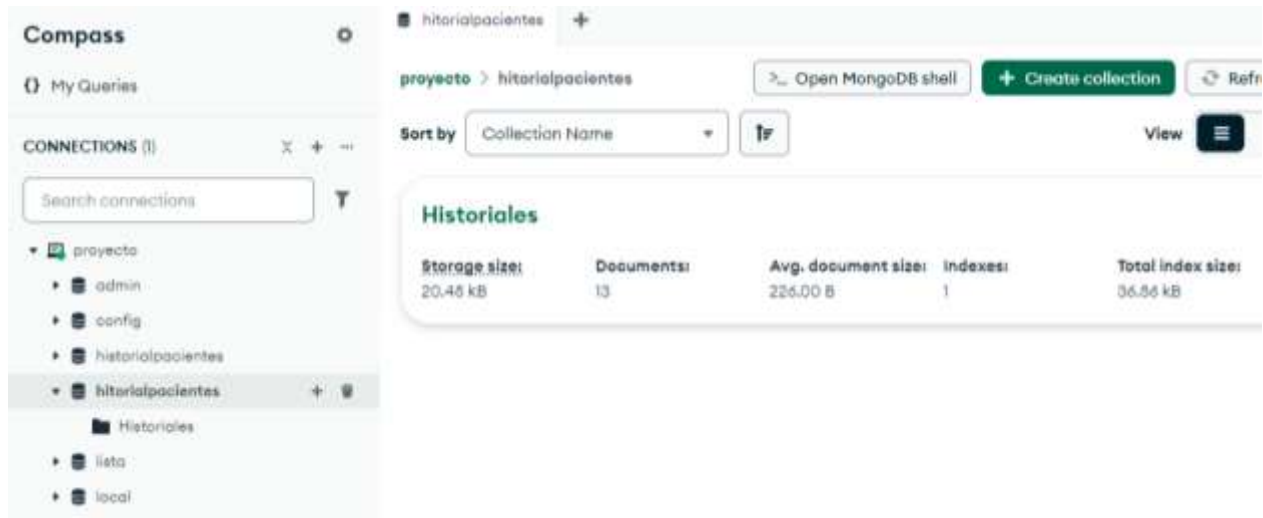


Figura 3.8 Descarga de gestor para la base de datos de historiales

En la figura 3.9 se observa la lista de los diferentes pacientes que se agregaron para realizar pruebas.

```
_id: ObjectId('67213381ca73b26ffe9353ad')
nombrePaciente: "lorena"
fecha: 2024-10-29T00:00:00.000+00:00
archivoECG: "historial1737486396184prueba (1).csv"
ssn: "123456789"
pacientId: "19940192"
age: 35
curp: "PIHL890913MDFQRR06"
phone: "2761034120"
gender: "F"
__v: 0
```

Figura 3.9 Base de datos de historiales médicos

3.4 Fase 2: Desarrollo de la página web

3.4.1 Etapa 1: Herramientas de desarrollo

- Visual Studio Code (desarrollo del código).
- MongoDB (gestión de la base de datos).

En esta etapa de la segunda fase se seleccionan los entornos de desarrollo para la página web de historiales médicos, así como las tecnologías necesarias para garantizar una captura de datos eficiente. Además, se busca proporcionar al usuario una experiencia práctica y sencilla al visualizar y gestionar cada uno de los módulos de la página web.

Sprint planning reunión

Durante el desarrollo de este sprint, se llevó a cabo la etapa de análisis para la elección de cada una de las tecnologías y entornos de desarrollo para la creación de la página web.

Objetivo: Establecer la funcionalidad básica del sitio y asegurar la integración de todos los componentes.

Descripción: En la reunión de planificación del sprint, se dedicará a examinar detalladamente el entorno de programación para el desarrollo adecuado e intuitivo de la página web.

Desglose de actividades

En la tabla 3.7 se detalla el desglose de actividades donde se realizó la revisión de la interfaz y algunas correcciones, la evaluación de los diferentes módulos y como implementarlos.

Descripción	Tareas	Horas
Analizar herramientas a utilizar	Revisión de la interfaz	3 horas
	Corrección de errores	4 horas
	Evaluación de los módulos	3 horas
	Revisión y análisis del diseño para su implementación	3 horas

Tabla 3.7 Actividades Scrum Team

Historias de usuario

Al utilizar esta página web, se busca que el usuario pueda registrar los historiales médicos de las pacientes de manera sencilla y práctica. El objetivo es solucionar la pérdida de tiempo en la gestión de información y permitir una respuesta oportuna en situaciones críticas para los médicos. Además, se pretende facilitar la detección de patologías a partir de las señales de los ECG, de una forma cómoda y accesible.

- **Presentación del diseño:** se preparó una presentación visual del diseño de la interfaz del sitio web, enfocada en la usabilidad y facilidad de acceso para los usuarios.
- **Creación de wireframes:** se utilizaron herramientas de diseño para crear wireframes de las páginas y los elementos del sitio web, asegurando una estructura clara y eficiente.
- **Iteraciones de diseño:** considerando los comentarios y sugerencias obtenidas durante las sesiones de revisión interna, se realizaron iteraciones en el diseño de la interfaz para mejorar su funcionalidad y experiencia de usuario.
- **Scrum Master:** durante esta reunión, se aprobaron y revisaron los requerimientos del proyecto, garantizando que todos los aspectos estuvieran alineados con los objetivos.

Product backlog

Entrega del Product Backlog: diseño de Interfaces de página web para generar historiales médicos. En la tabla 3.8 se detalla la prioridad y complejidad que se tubo al realizar el diseño de interfaz y los módulos que se crearon para la página web de los historiales médicos.

Descripción	Tarea	Prioridad	Complejidad	No. Sprint	Días
Diseño de interfaz de Inicio intuitivo, facilitando al usuario la captura de los archivos clínicos	Diseñar la interfaz de inicio, con secciones fáciles de comprender y que permitan al usuario practicidad en	5	4	1	3

	su navegación.				
Diseño de módulos de fácil acceso para identificar las diferentes funciones con las que cuenta la página web	Diseñar 4 módulos para que el usuario pueda interactuar con la página web	5	5	1	3
Diseño de funciones que permita al usuario capturar la información para generar un historial clínico por medio de java script	Diseñar funciones en java script como la captura de datos y poder guardar esta información en una base de datos	5	5	1	2
Recabar la información que se necesita para poder crear un historial medico	Investigar la información necesaria que se requiere en historial médico con respecto a mujeres embarazadas	5	4	1	1
Diseño del apartado en el cual se pueden cargar archivos CSV para las señales de	Diseñar un apartado donde el usuario observe la localización del ECG	5	5	1	2

imágenes de ECG					
Diseño de un apartado para poder mostrar los historiales clínicos de cada paciente mediante el id del paciente	Buscar un diseño sencillo donde la información sea lo más clara posible y que muestre lo más relevante del paciente	5	5	1	3
Desarrollo de optimización dentro del sitio web		5	5	1	2
Diseño de los diferentes módulos para el fácil acceso del usuario y pueda visualizar archivos clínicos, historiales e imágenes CSV, así como un login de acceso para usuarios y administradores	Diseñar los módulos que mostraran la información que el Scrum Master pidió	5	5	1	3

Tabla 3.8 Product Backlog diseño

En la fase actual del Product Backlog, el enfoque principal está en el desarrollo de las interfaces para la página web destinada a la creación y gestión de historiales médicos de mujeres embarazadas. Las historias de usuario y las tareas asociadas se han adaptado cuidadosamente para satisfacer las necesidades de los profesionales de la salud, asegurando que el sistema sea funcional y fácil de usar.

Sprint backlog

En términos generales, el enfoque del Sprint se centra en la optimización del diseño de la página web con el fin de mejorar la experiencia de los usuarios. Se busca facilitar el acceso a los diferentes módulos asegurando que la plataforma sea intuitiva y eficaz para el usuario, así mismo se busca que la página web sea fácil de utilizar que el usuario pueda actualizar y modificar los datos capturados en nuestro historial creado, de tal forma que pueda modificar si es necesario.

En la tabla 3.9 se puede apreciar el proceso del diseño de la página web, con el fin de crear una navegación intuitiva la cual permita acceder fácilmente.

Producto	Pendiente	En progreso	Finalizado
Sprint 1		Diseñar una navegación intuitiva que permita a los usuarios acceder fácilmente a los diferentes módulos de la página	Crear diseños de maquetación de la página web, captura de datos, búsqueda de historiales, login.
Diseñar la maquetación de la página web de manera que cumpla con los requisitos necesarios			

Tabla 3.9 Sprint backlog

3.4.2 Etapa 2: Gestión de base de datos

Sprint planning

Durante este sprint se centró proceso de creación, organización, mantenimiento y control de la información almacenada en la base de datos.

Objetivos: Es utilizar el gestor MongoDB para desarrollar las funcionalidades requeridas para la estructura de la base de datos, incluyendo tablas, colecciones, esquemas, en bases de datos No SQL.

Descripción: Poder registrar y subir datos en función de las necesidades del sistema, tener una mejor optimización, mejorar el rendimiento de consultas y operaciones.

Además, se pondrá énfasis en asegurar que los datos puedan ser restaurados en caso de fallos del sistema o pérdida de información y supervisar el rendimiento de la base de datos.

Desglose de actividades

En la tabla 3.10 se puede observar el desglose de actividades en el cual se realizó la revisión de la base de datos, su gestión y cada uno de los campos para la estructura de la captura de los historiales médicos.

Descripción	Tarea	Horas
Gestión de la base de datos	Revisión de la gestión de la base de datos	4
	Diseñar los campos en la base de datos	4
	Definición de la estructura y modificación de la base de datos	4
	Creación de consultas en la base de datos	4

Tabla 3.10 Actividades Scrum Team

Historias de usuario

- Crear y configurar la base de datos para garantizar que el sistema pueda almacenar la información de los pacientes de manera segura y organizada.
- Gestionar permisos de acceso para los usuarios para asegurar que solo personal autorizado pueda visualizar o modificar los datos.

- Buscar el historial médico de un paciente por su nombre o ID para acceder rápidamente a la información
- Permitir almacenar diferentes tipos de datos médicos de manera eficiente.

Product backlog

En la tabla 3.11 se muestra el desarrollo y entrega del Product Backlog este se enfoca en la gestión adecuada del gestor de base de datos para el historial médico. Las historias de usuario y las tareas asociadas describen las funcionalidades específicas que deben ser implementadas.

Descripción	Tarea	Prioridad	Complejidad	No. Sprint	Días
Implementación del gestor adecuado para la base de datos	Crear los campos necesarios para la captura de los historiales médicos.	5	5	1	5
Crear un módulo para la visualización de los historiales médicos y la información guardada	Diseñar una tarjeta para la visualización de los datos requeridos de cada paciente	4	5	1	5
Crear un diseño sencillo pero entendible al usuario	El diseño de la muestra de los datos del paciente debe ser sencillo pero claro	4	5	1	5

Tabla 3.11 Product Backlog

Sprint backlog

El enfoque del Sprint se centra en la optimización de la base de datos con el fin de mejorar la experiencia de los usuarios. Facilitar la visualización de los datos de cada paciente de manera personalizada por medio del ID asignado a cada paciente, esto mediante el uso del módulo de búsqueda en la página web, para que el usuario pueda visualizarlo de manera fácil, como se detalla en la tabla 3.12.

Producto	Pendiente	En progreso	Finalizado
Sprint 1		Tener un gestor de base de datos eficaz en el cual se puedan listar los pacientes capturados	Mostrar las tarjetas de búsqueda con la información necesaria del paciente.
Gestionar la base de datos para la visualización ordenada de los pacientes registrados y su correspondiente historial médico			

Tabla 3.12 Sprint backlog

3.5 Fase 3: Integración de Inteligencia Artificial

3.5.1 Etapa 1: Procesamiento de señales de ECG.

Desarrollo de una red neuronal en Python

En esta etapa del proyecto, se implementa una red neuronal para el procesamiento de imágenes de electrocardiogramas de mujeres embarazadas. Este desarrollo se lleva a cabo utilizando **Google Colab**, una plataforma que permite ejecutar y entrenar modelos de aprendizaje profundo aprovechando recursos de hardware. La red neuronal se encarga de procesar datos provenientes de archivos **CSV** que contienen información de los electrocardiogramas. Estos archivos son generados a partir de los estudios realizados y se suben al historial médico de cada paciente en la página web del sistema.

Sprint Planning reunión

Objetivo: Proporcionar resultados facilitando la interpretación de datos para especialistas.

Descripción

3.5.2 Pre procesamiento de datos

Los archivos CSV se cargan en Google Colab, donde se realiza el pre procesamiento de los datos, como la normalización y la transformación de las señales.

Se utiliza una biblioteca como Matplotlib para convertir los datos en gráficos de alta calidad que representen el electrocardiograma.

3.5.3 Diseño de la red neuronal

Se implementa una red neuronal convolucional (CNN) utilizando TensorFlow. Este tipo de red es ideal para procesar imágenes y extraer características clave, en este caso se nombró Modelo Convolucional 32 con capas que se utilizan para la extracción de características de las imágenes, fundamentales en redes neuronales convolucionales (CNN) y se encargan de extraer características de las imágenes, como bordes, texturas, formas y patrones.

En la tabla 3.13 se encuentra el desglose de actividades, en el cual se hace un análisis de las librerías aptas para entrenar la red neuronal y hacer diferentes iteraciones con el optimizador más adecuado, el cual permita el mejor procesamiento de las imágenes.

Desglose de actividades

Descripción	Tareas	Días	Horas
Analizar las herramientas para crear la red neuronal	Revisión de librerías a utilizar	1	4
	Iteraciones de diferentes optimizadores para el mejor procesamiento de las imágenes	3 días	4
	Detectar el formato ideal para el procesamiento de imágenes	2 días	4
	Revisión y análisis de la red neuronal funcional	3 días	4

Tabla 3.13 Actividades Scrum Team

Historias de usuario

- Subir archivos CSV de electrocardiogramas al historial médico de los pacientes para garantizar que los datos estén disponibles para su procesamiento.
- Confirmar que las imágenes de los electrocardiogramas generadas por el sistema se asocien al paciente correcto para evitar errores en la asignación de información.
- Implementar una API que conecte la página web con la red neuronal para que las imágenes procesadas estén disponibles en el historial médico de manera automática.

Product backlog

El desarrollo de esta etapa nos ayuda a verificar las herramientas necesarias para el procesamiento de imágenes de la red neuronal, el uso de las librerías adecuadas y analizar el uso de optimizadores adecuados para la red neuronal. Las historias de usuario y las tareas asociadas describen las funcionalidades específicas que deben ser implementadas, así como el procesamiento de audio y el proceso de conversión de la imagen en la red neuronal para mostrar la imagen del electrocardiograma de la madre.

En la tabla 3.14 se puede apreciar el número de días que se llevaron para el análisis correcto del entorno más adecuado para trabajar la red neuronal, el cual en este caso nos ayudó a procesar eficientemente las imágenes de señales de ECG.

Descripción	Tarea	Prioridad	Complejidad	No. Sprint	Días
Implementación del entorno adecuado para la red neuronal	Analizar el entorno adecuado para trabajar el procesamiento de imágenes en la red.	5	5	1	5
Adecuar las capas necesarias para el procesamiento de la red neuronal	Estudiar las diferentes capas neuronales y las librerías adecuadas para el correcto	5	5	1	5

	funcionamiento de la red neuronal.						
Crear las diferentes iteraciones con los optimizadores más adecuados para el procesamiento	Guardar las diferentes pruebas que se realizan con los diferentes optimizadores para encontrar el más estable.	5	5	1	6		

Tabla 3.14 Product Backlog

Sprint backlog

En la tabla 3.25 se detalla el enfoque del Sprint que se centra en la optimización de las diferentes iteraciones en la red neuronal para su mejor procesamiento y facilitar la visualización de las imágenes.

Producto	Pendiente	En progreso	Finalizado
Sprint 1		Probar los diferentes optimizadores para el procesamiento de las imágenes	Mostrar las gráficas en las cuales se muestran las épocas y la exactitud del análisis de la red.
Hacer las pruebas necesarias para las diferentes iteraciones con los optimizadores adecuados para el funcionamiento de la red neuronal			

Tabla 3.15 Sprint backlog

3.5.4 Etapa 2: Interpretación de datos

Interpretación de datos en el sistema de historiales médicos

La interpretación de datos es una etapa crucial dentro del sistema, ya que permite transformar los datos procesados en información significativa que pueda ser utilizada por médicos para la toma de decisiones clínicas. Este proceso combina herramientas de inteligencia artificial, procesamiento de señales y visualización de datos.

Flujo para la interpretación de datos

1. Recolección de datos

- Los datos iniciales, como señales de electrocardiogramas en formato CSV y datos clínicos básicos de los pacientes, son ingresados al sistema.
- Estos datos son almacenados en la base de datos MongoDB y vinculados al historial médico del paciente.
- Se realizaron las pruebas pertinentes en Postman para verificar que se subieron correctamente los archivos CSV.

2. Procesamiento previo

- Los archivos CSV de los electrocardiogramas son transformados en audio y posteriormente a un espectrograma mediante herramientas como Matplotlib en Google Colab.
- El sistema realiza un pre procesamiento, como filtrado de ruidos o normalización de señales, para asegurar que los datos sean claros y utilizables por la red neuronal.

3. Análisis mediante la red neuronal

- La red neuronal convolucional (CNN) procesa las imágenes generadas a partir de los datos del electrocardiograma.

4. Integración de resultados con el historial médico

- Los resultados interpretados por la red neuronal son almacenados junto con las imágenes y los datos clínicos del paciente.

3.6 Fase 4: Implementación y Alojamiento

- **Implementación del sitio web:**
 - Alojamiento en un servicio de hosting gratuito.
- **Objetivo:** Asegurar la disponibilidad y el acceso del sitio.

Sprint backlog

Las visualizaciones integrales de las tareas relacionadas con la página web y los historiales clínicos son fundamentales en el sprint actual. El equipo de desarrollo se encargó de gestionar y dar seguimiento a estas tareas, asegurando que la implementación de la plataforma fuera eficiente y cumpliera con los requisitos establecidos. La correcta visualización de los historiales médicos y la carga de archivos CSV es importante para proporcionar una herramienta efectiva al usuario.

Una vez revisados los requerimientos, se procedió a seleccionar un hosting adecuado para el alojamiento de la página web. En este caso, se optó por utilizar GitHub, lo que permitió subir la página web sin inconvenientes. Además, se configuró la colaboración con otro usuario, facilitando el trabajo en equipo y asegurando que, en el futuro, se puedan realizar mejoras y los cambios necesarios de manera ágil y organizada, según lo requiera el proyecto.

Sprint planning

El enfoque principal estará en la implementación y las pruebas de la página para la creación de historiales médicos, se describen algunas actividades comunes que podrían llevarse a cabo durante este sprint

Objetivo: Es tener una página web funcional, libre de errores y preparada para su uso o para futuros sprints de mejora y optimización.

Descripción: En el transcurso del Sprint 4, el equipo de desarrollo se dedicará a traducir las funcionalidades específicas previamente delineadas en el Product Backlog en elementos tangibles y concretos para la página web los médicos y expertos de la salud. Este proceso abarcará desde la implementación de código hasta la realización de

pruebas para garantizar el correcto funcionamiento de las características desarrolladas, consolidando así el avance significativo hacia la culminación exitosa del proyecto.

Scrum Team

1. Página web funcional: la página web debe estar completamente implementada, con todas las funcionalidades requeridas según los requisitos establecidos. Los usuarios deben poder entrar con facilidad, ver y revisar toda la información disponible sobre los historiales médicos.
2. Pruebas exitosas: se deben realizar pruebas tanto la página web para verificar su correcto funcionamiento, los errores deben ser corregidos para garantizar la calidad del software.
3. Estabilidad y usabilidad: el sitio web debe ser estable y fácil de usar. Se deben realizar mejoras y ajustes según sea necesario para optimizar la experiencia del usuario y asegurar que el software sea intuitivo y fácil de navegar.

Plan de Sprint

En la tabla 3.16 se puede observar la tabla de actividades del Scrum Team durante el sprint, en la que se implementaron las sugerencias para mejorar la página web y realizar las pruebas necesarias para ver la conexión de la base de datos y su correcto funcionamiento.

Descripción	Tareas	Horas
Implementar todas las funcionalidades planificadas para la página web siguiendo los requisitos y las especificaciones definidas	Análisis de la interfaz	5
Integrar las diferentes partes de la página web para asegurar su correcto funcionamiento.	Realización de pruebas	5

Realizar pruebas de aceptación para validar que la página web cumple con los requisitos del usuario y funcionan según lo esperado.	Verificar pruebas finales de la página web.	5
Corregir errores y realizar mejoras en la página web, en base a los resultados de las pruebas y las retroalimentaciones recibidas.	Realizar las correcciones sugeridas.	5

Tabla 3.16 Plan de sprint de actividades.

Conclusión y compromiso: Es fundamental considerar la escalabilidad del sistema, asegurándose de que pueda manejar un crecimiento sostenido en el volumen de datos. Asimismo, la documentación detallada del código y la adopción de patrones de diseño eficientes contribuyen significativamente a la mantenibilidad a largo plazo del software, facilitando futuras actualizaciones y mejoras.

Historias de usuario

- Pruebas unitarias: realizar pruebas a nivel de módulo para verificar su correcta funcionalidad.
- Pruebas de integración: probar la interacción entre los diferentes módulos de la página web para asegurar su correcta integración y funcionamiento conjunto.
- Pruebas de rendimiento: evaluar el rendimiento estas pruebas buscan identificar posibles errores, optimizar el rendimiento y garantizar una buena experiencia de uso.
- Ajustes y correcciones: se realizaron los ajustes y correcciones necesarios en la página web con base en los resultados de las pruebas realizadas. Esto implica solucionar errores, mejorar el rendimiento y optimizar la funcionalidad de acuerdo con los resultados encontrados.
- Preparación para la implementación: preparar los entornos de producción y pruebas para la implementación de la página web. Esto puede incluir

configuraciones de servidores, utilizar archivos y asegurar los recursos necesarios para la puesta en marcha.

Sprint backlog

En la tabla 3.17 se observa el Sprint Backlog la cual exhibe un registro de las actividades de los usuarios elegidas para el Sprint, junto con las acciones requeridas para llevar a cabo cada tarea.

Producto	Pendiente	En progreso	Finalizado
Sprint 4			Establecer el entorno de pruebas y organizar los datos.
Objetivo de sprint: consiste en contar con una página web, sin errores, lista para ser usada o para futuras fases de mejora.			Crear y llevar a cabo casos de prueba.
			Documentar y registrar los resultados obtenidos.

Tabla 3.17 Sprint backlog

CAPÍTULO IV
ANÁLISIS Y DISCUSIÓN DE
RESULTADOS

4.1 Introducción

En el siguiente capítulo se presentan los resultados obtenidos del desarrollo del proyecto que tiene por nombre “Sistemas digital de gestión de historiales médicos obstétricos”

4.2 Implantación y creación de la página web

En el presente proyecto se utilizó un hosting gratuito que permitió subir la página web manteniendo su diseño en este caso se utilizó GitHub. Lo primero que se realizó fue la búsqueda del hosting GitHub desde el buscador posteriormente para ingresar se llevó a cabo el registro mediante un correo electrónico.

En la figura 4.1 se muestra el registro del hosting que se eligió para subir la página web.

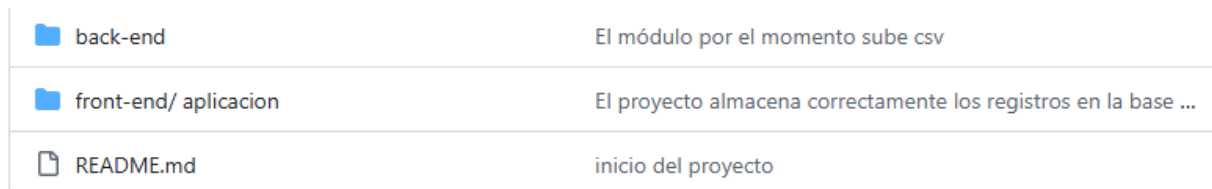


Figura 4.1 Registro en GitHub

Posteriormente se presentaron aquellas verificaciones que el hosting necesita, la creación de un perfil para subir con seguridad el proyecto, una vez iniciado se mostró la pantalla principal, se sube la carpeta del proyecto de la página web, con la opción de

compartido para modificaciones futuras, se puede descargar o bien clonar el repositorio del proyecto para posibles cambios.

En la figura 4.2 se muestra el proyecto que fue subido al hosting de GitHub, se pueden guardar cambios para evitar perder los cambios realizados y tener un respaldo seguro.



back-end	El módulo por el momento sube csv
front-end/ aplicacion	El proyecto almacena correctamente los registros en la base ...
README.md	inicio del proyecto

Figura 4.2 Archivos subidos al hosting

El resultado final presenta los diferentes módulos en los que se muestra el inicio de la página que es el login, la captura del historial médico, búsqueda del historial y el historial del paciente. Estos elementos incluyen:

Interfaces: muestra los diferentes módulos, con un diseño sencillo implementado. El modulo principal muestra el inicio de sesión para acceder como usuario o administrador, en la parte superior se encuentra un menú de cuatro opciones que son enlaces a los diferentes módulos para la captura de la historia clínica y visualizar los datos.

En la figura 4.3 se observa el modulo principal de la página, el login de inicio para acceder a la página, como usuario o como administrador, el cual tiene la aprobación para modificar la base del historial médico o realizar cambios en la captura de datos de cada paciente.

The screenshot shows a web interface with a top navigation bar containing the links: INICIO, BUSQUEDA, HISTORIAL PACIENTE, and LOGIN. Below this bar, the page is titled "Iniciar Sesión / Registro". The form includes a "Usuario:" label above a text input field with the placeholder "Ingresa tu usuario". Below this is a "Contraseña:" label above a text input field with the placeholder "Ingresa tu contraseña". To the right of the password field is a dropdown menu labeled "Rol:" with "Usuario" selected. At the bottom of the form are two red buttons: "Ingresar" and "Registrar".

Figura 4.3 Inicio de sesión de la página

En la figura 4.4 se aprecia el modulo en el cual se realiza la captura de datos para el historial y el campo en el cual se puede subir el archivo CSV y el electrocardiograma.

The screenshot shows a web interface with a top navigation bar containing the links: INICIO, BUSQUEDA, HISTORIAL PACIENTE, and LOGIN. Below this bar, the page is titled "Historial para Electrocardiograma". The form includes a "Nombre del Paciente:" label above a text input field. Below this is a "Fecha:" label above a date input field with the placeholder "dd / mm / aaaa" and a calendar icon. Below the date field is a "Número de Seguro Social (SSN):" label above a text input field. Below the SSN field is an "ID de Paciente:" label above a text input field. Below the ID field is an "Edad:" label above a text input field. Below the age field is a "CURP:" label above a text input field. Below the CURP field is a "Teléfono:" label above a text input field. Below the phone field is a "Sexo:" label above a dropdown menu with "Selecciona" and a downward arrow. Below the sex dropdown is an "Archivo ECG (CSV):" label above a text input field with the placeholder "Elegir archivo". Below the ECG field is a red button labeled "Guardar Datos".

Figura 4.4 Apartado para captura de datos de pacientes

En la figura 4.5 se detalla como se realizó la búsqueda de pacientes insertando el Id generado en la base de datos, con esta acción se puede buscar un paciente en específico y modificar datos.

The screenshot shows a web application interface for patient search. At the top, there is a navigation bar with four links: INICIO, BUSQUEDA (highlighted in red), HISTORIAL PACIENTE, and LOGIN. Below the navigation bar, the main heading is "Búsqueda de historial Médico". Under this heading, there is a text input field containing the number "291189". To the right of the input field is a red button labeled "Buscar". Below the search section, the heading "Información del Paciente" is followed by a list of patient details: "Nombre: Maria Fernanda", "ID: 291189", "Edad: 25", "Fecha: 2011-02-01T00:00:00.000Z", "Telefono: 2761030989", "Genero: F", and "Historia Médica: default.ccv".

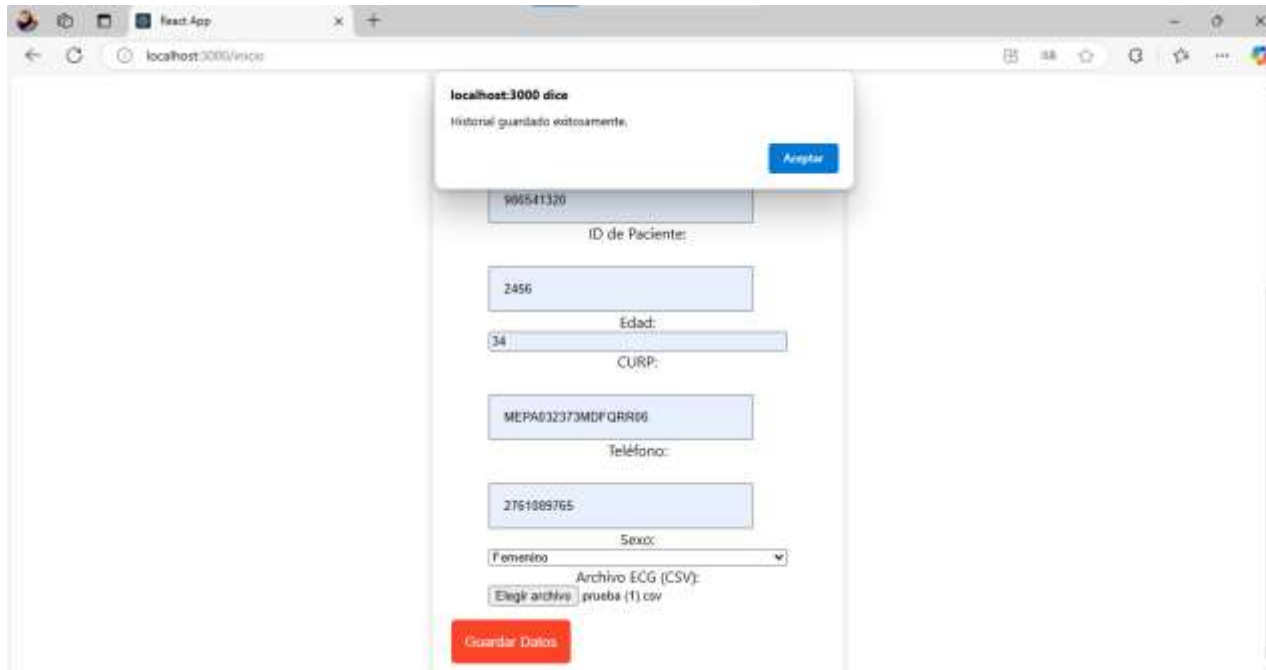
Figura 4.5 Módulo para búsqueda de pacientes

En la figura 4.6 se muestra como se exportan los historiales en Excel, o bien un historial en específico.

The screenshot shows a web application interface for patient history. At the top, there is a navigation bar with four links: INICIO, BUSQUEDA, HISTORIAL PACIENTE (highlighted in red), and LOGIN. Below the navigation bar, the main heading is "Historial de Pacientes". Under this heading, there is a table with the following header: "Nombre ID Acciones". Below the header, there is a red button labeled "Exportar a Excel".

Figura 4.6 Módulo para descargar el historial del paciente

En la figura 4.7 se aprecia que al capturar los datos del paciente y el archivo CSV correspondiente se guardan correctamente los datos y se sube a la base de datos.



The screenshot shows a web browser window with the address bar displaying 'localhost:3000/escio'. The main content area features a form for patient data. At the top, a white modal box displays the message 'localhost:3000 dice' and 'Historial guardado exitosamente.' with a blue 'Aceptar' button. Below this, the form contains several input fields: 'ID de Paciente' (905541320), 'Edad' (2456), 'CURP' (34), 'Teléfono' (MEPA032373MDFQRR06), and 'Sexo' (Femenino). There is also a section for 'Archivo ECG (CSV)' with a file selection button 'Elegir archivo' and a preview 'prueba (1).csv'. A red 'Guardar Datos' button is at the bottom of the form.

Figura 4.7 Carga de archivo CSV en la página

4.3 Pruebas Técnicas

4.3.1 Validación de datos

Se realizaron pruebas del funcionamiento de la página en la API de Postman para verificar que los archivos en formato CSV se subieran correctamente y que no se presentaran errores, así como también la muestra de los historiales clínicos en la base de datos correctamente.

En la figura 4.8 se aprecia una prueba básica para comprobar que la base de datos se conectó correctamente a la página web.



Figura 4.8 Prueba en Postman

En la Figura 4.9 se presenta un listado de los pacientes de prueba registrados en su historial médico, junto con los datos correspondientes generados en el expediente de cada uno.

```

1 1
2   "status": "OK",
3   "historiales": [
4     {
5       "_id": "678fe9149fd3f8127e064ff1",
6       "nombrePaciente": "EVELIA HERNÁNDEZ CALZADA",
7       "fecha": "2025-01-21T00:00:00.000Z",
8       "archivoECG": "historial1737485908302prueba.csv",
9       "ssn": "987654321",
10      "pacientId": "291189",
11      "age": 37,
12      "curp": "PIHL890913MDFQRR06",
13      "phone": "2761034120",
14      "gender": "F",
15      "__v": 0

```

Figura 4.9 Prueba de listar historiales guardados

En la Figura 4.10 se muestra una prueba realizada para verificar el historial médico del paciente, asegurando que los datos se han capturado correctamente y que la conexión con la base de datos es exitosa

```
1  {
2    "status": "OK",
3    "articulo": {
4      "_id": "67213381ca73b26ffe9353ad",
5      "nombrePaciente": "lorena",
6      "fecha": "2024-10-29T00:00:00.000Z",
7      "archivoECG": "historial1737486396184prueba (1).csv",
8      "ssn": "123456789",
9      "patientId": "19940192",
10     "age": 35,
11     "curp": "PIHL890913MDFQRR06",
12     "phone": "2761034120",
13     "gender": "F",
14     "__v": 0
```

Figura 4.10 Prueba archivo CSV subido correctamente

4.3.2 Pruebas de Configuración

Se realizaron pruebas detalladas para validar la correcta configuración y visualización de la página web en la API. Esto incluyó la verificación de ser compatible con las tecnologías a usar con la Red Neuronal en Python.

4.3.3 Análisis Experimental

Se realizaron pruebas experimentales para evaluar la eficiencia y efectividad de características específicas de la página web, que incluyeron pruebas de usabilidad, análisis de tiempos de carga y recopilación de datos.

4.4 Red Neuronal (CNN)

4.4.1 Pruebas realizadas en una red neuronal para el procesamiento de imágenes de electrocardiogramas de madres embarazadas

Se realizaron pruebas en una red neuronal desarrollada en Google Colab, diseñada específicamente para el análisis de imágenes de electrocardiogramas de mujeres embarazadas. Para comenzar, se utilizó un conjunto de imágenes de prueba para

entrenar y validar una red neuronal de pequeña escala, centrada en el procesamiento de imágenes.

Cabe mencionar que todo esto se programó con respecto a adaptación de la red neuronal en la cual se introdujeron capas de entrada para las imágenes y poder después recibir los espectrogramas creados a partir de las señales no invasivas de Electrocardiogramas Fetales (NI-FECG), esto por medio de bloques en cada capa de convolución.

En la Figura 4.11 se muestran las pruebas realizadas para evaluar los diferentes optimizadores utilizados en la red neuronal, con el objetivo de analizar su desempeño en términos de estabilidad y precisión del modelo.

```
def build_model(num_classes):  
    model = tf.keras.Sequential([  
        tf.keras.layers.InputLayer(input_shape=(5998, 32, 1)),  
  
        tf.keras.layers.Conv2D(filters=8, kernel_size=(3, 3), activation=None),  
        tf.keras.layers.BatchNormalization(),  
        tf.keras.layers.ReLU(), # Changed ReLu to ReLU  
  
        tf.keras.layers.Conv2D(filters=16, kernel_size=(3, 3), activation=None),  
        tf.keras.layers.Dropout(0.25),  
        tf.keras.layers.MaxPooling2D(pool_size=(2, 2), strides=2),  
        tf.keras.layers.ReLU(), # Changed ReLu to ReLU  
  
        tf.keras.layers.Conv2D(filters=32, kernel_size=(3, 3), activation=None),  
        tf.keras.layers.Dropout(0.25),  
        tf.keras.layers.MaxPooling2D(pool_size=(2, 2), strides=2), # Changed tf.keras.layers to tf.keras.layers  
        tf.keras.layers.ReLU(), # Changed ReLu to ReLU  
  
        tf.keras.layers.Flatten(),  
        tf.keras.layers.Dense(128, activation=None),  
        tf.keras.layers.ReLU(), # Changed ReLu to ReLU  
        tf.keras.layers.Dense(64, activation=None),  
        tf.keras.layers.ReLU(), # Changed ReLu to ReLU  
  
        tf.keras.layers.Dense(num_classes, activation='softmax') # Changed num_clases to num_classes
```

Figura 4.11 Capas de red neuronal para el procesamiento de las señales

Los resultados obtenidos incluyen métricas detalladas de exactitud, así como gráficas que ilustran el comportamiento del modelo a lo largo de las épocas de entrenamiento para cada optimizador. Además, se generaron gráficas de las pérdidas durante el entrenamiento, permitiendo identificar de manera precisa la mejor configuración y optimización para el entrenamiento de la red neuronal.

Estas gráficas de precisión y pérdidas muestran claramente cuál de los optimizadores utilizados ofrece un desempeño más estable y consistente para el

procesamiento de imágenes, proporcionando así una base sólida para seleccionar el mejor enfoque en futuras implementaciones.

En la figura 4.12 se muestra la gráfica del optimizador Adam, en la que se puede ver de una manera precisa el nivel de exactitud en la red neuronal, la cual no es estable para el entrenamiento de la red.

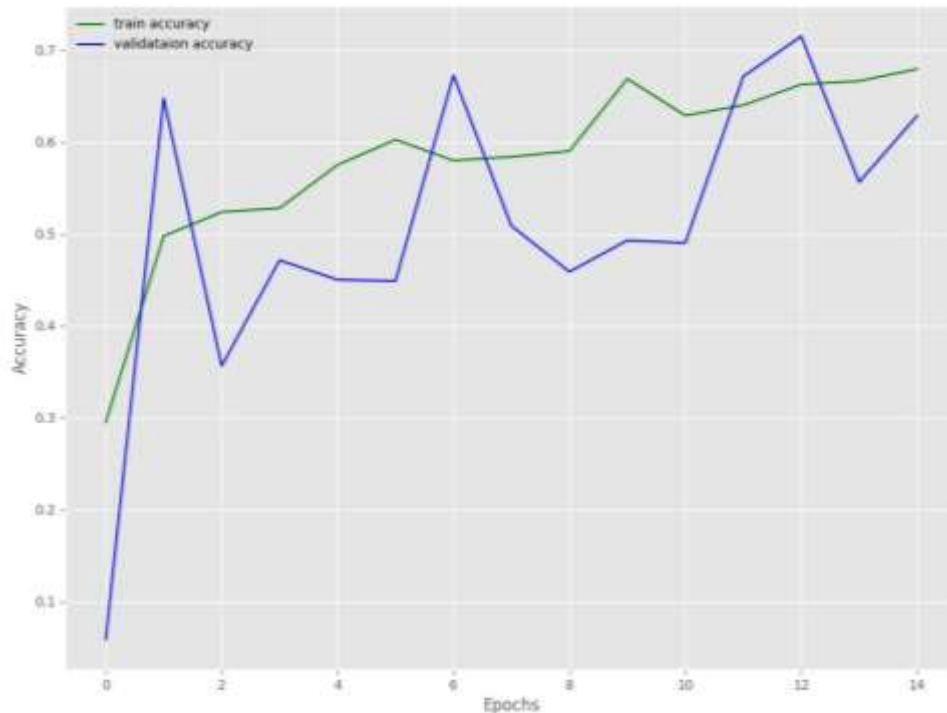


Figura 4.12 Accuracy de optimizador Adam

En la figura 4.13 se muestra la perdida de el optimizador Adam, la cual se muestra inestable para la red neuronal.

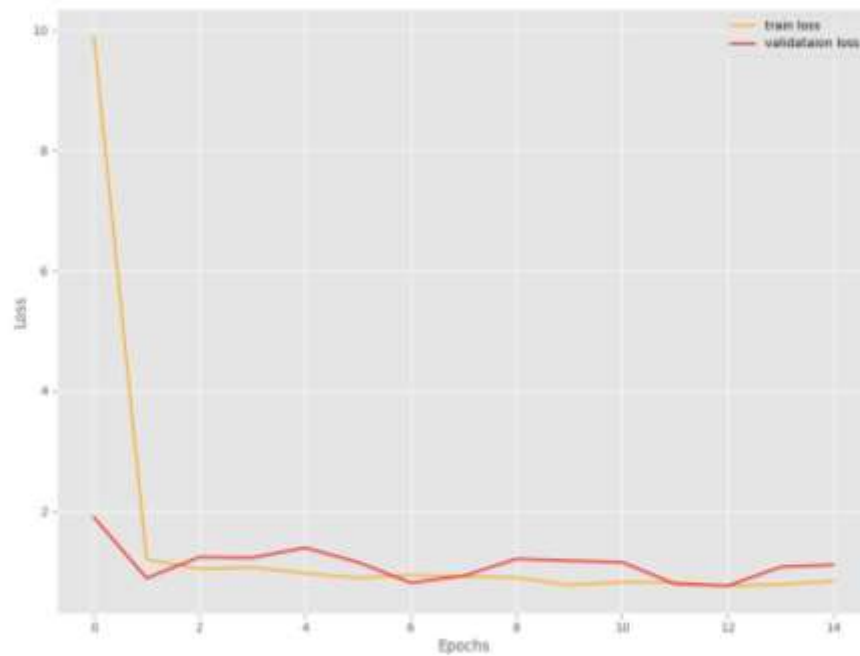


Figura 4.13 Loss optimizador Adam

En la figura 4.14 vemos el optimizador SGD, que muestra un nivel de exactitud mejor pero aun no apto para la red neuronal.

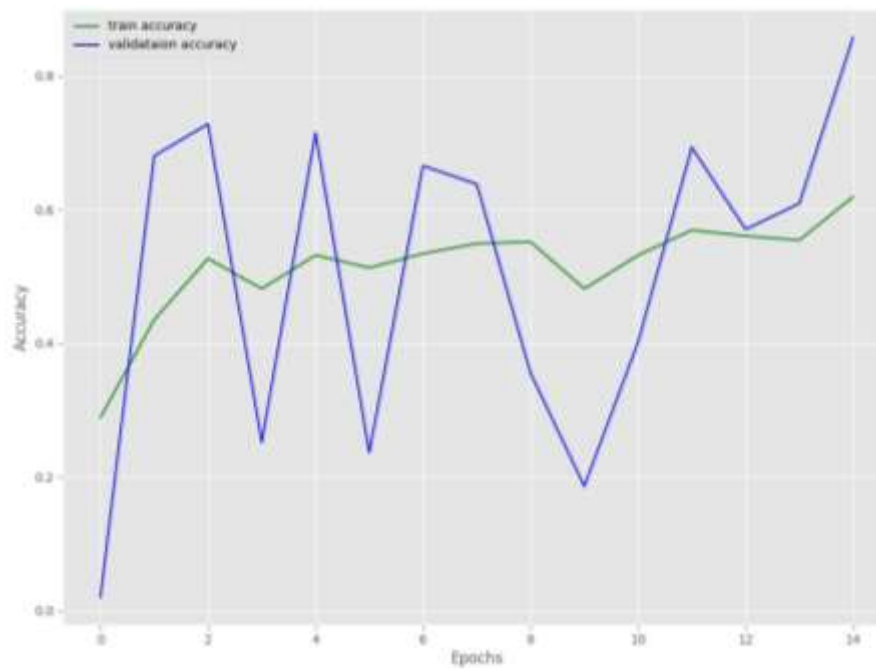


Figura 4.14 Accuracy de optimizador SGD

En la figura 4.15 se detalla la pérdida del optimizador SGD en el cual se muestra una inestabilidad mayor no apta para la red neuronal.

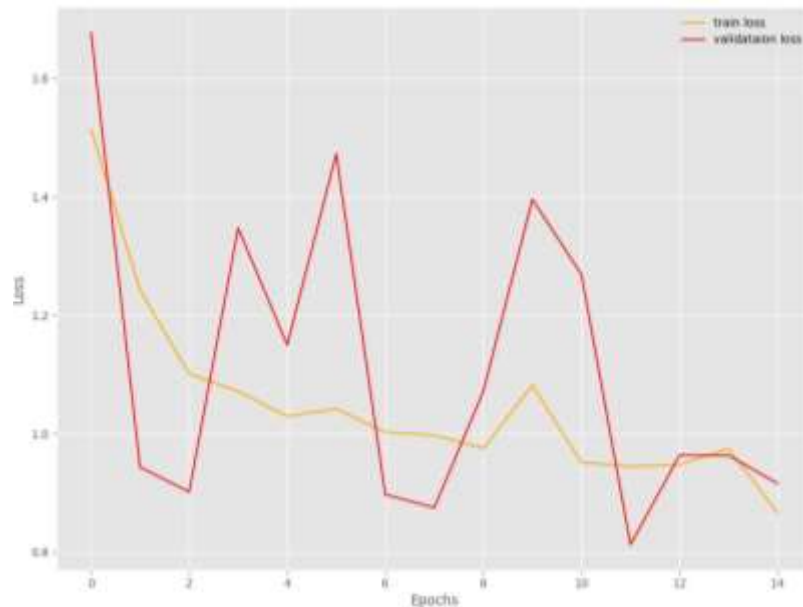


Figura 4.15 Loss de optimizador SGD

En la figura 4.16 se puede apreciar la exactitud del optimizador Adamax, la cual es muy inestable y no apta para la red neuronal.

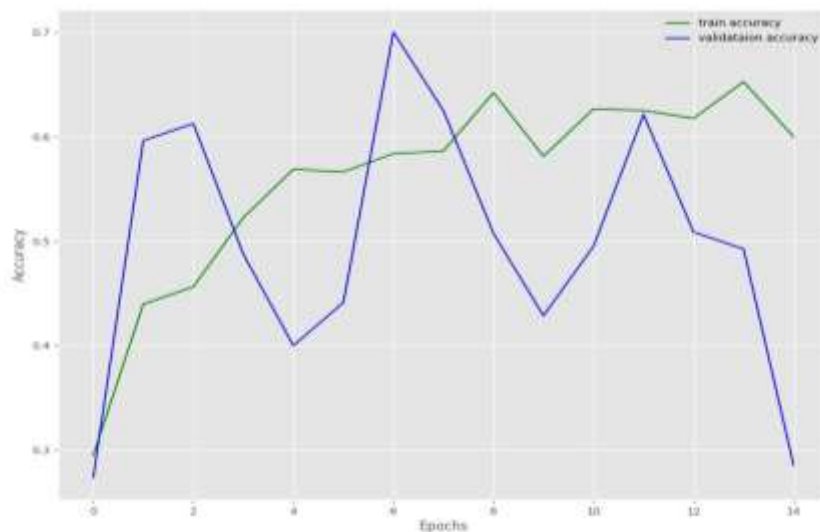


Figura 4.16 Accuracy de optimizador Adamax

La figura 4.17 se observa la pérdida del optimizador Adamax, se puede apreciar que es menos, pero aun no apta para el entrenamiento de la red neuronal.

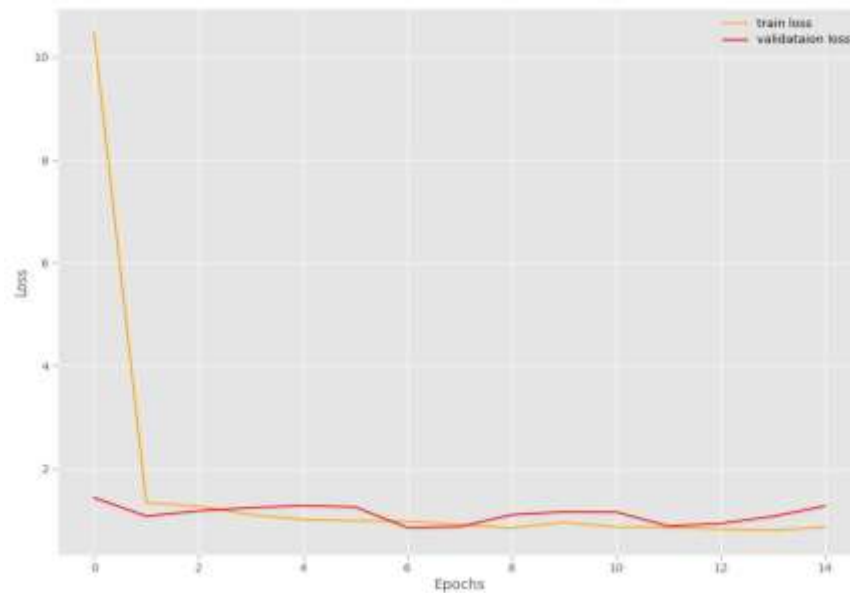


Figura 4.17 Loss optimizador Adamax

Y por último en la figura 4.18 con el siguiente optimizador que fue RMSprop se llegó a la conclusión que es el más estable para el aprendizaje de la red neuronal en el procesamiento de las imágenes, el cual muestra en sus graficas la menor perdida y la mejor exactitud en su aprendizaje.

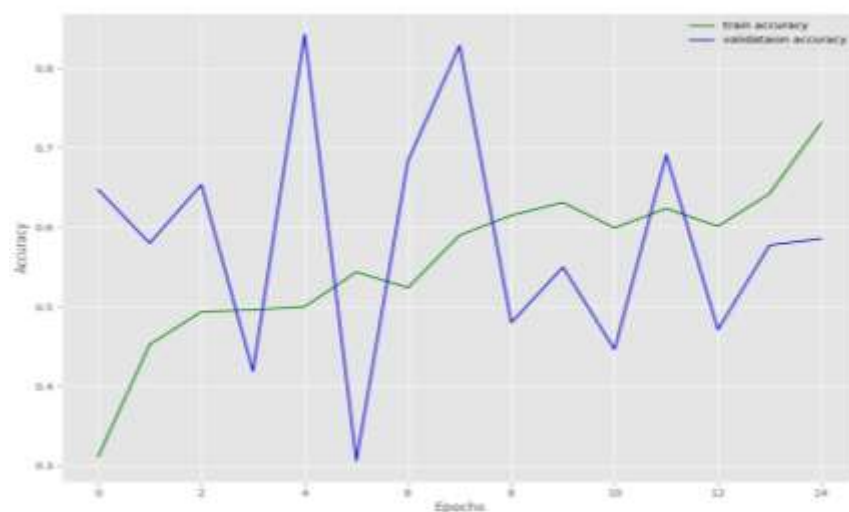


Figura 4.18 Accuracy optimizador RMSprop

En la figura 4.19 se observa la perdida del optimizador RMSprop el cual en su grafica muestra la menor perdida y el mas apto para el entrenamiento de la red.

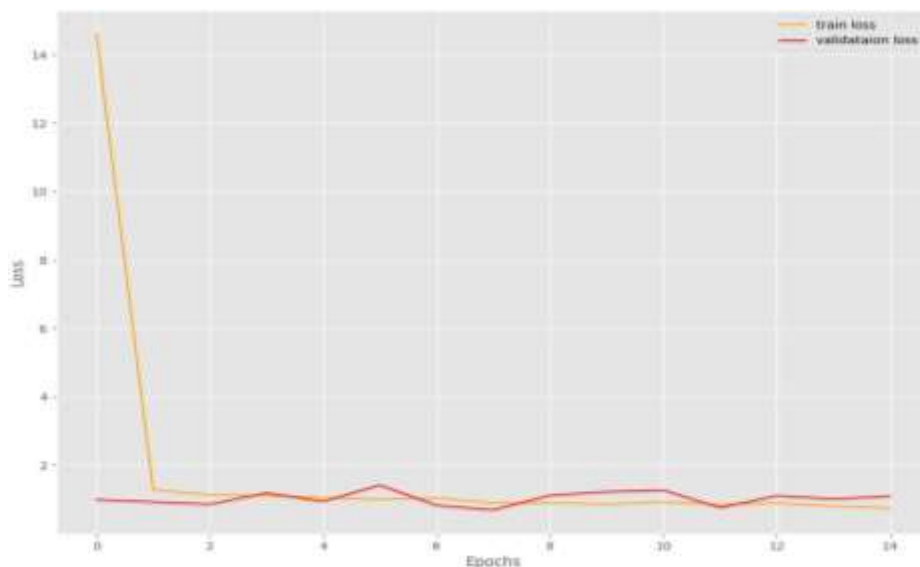


Figura 4.19 Loss de optimizador RMSprop

4.5 Procesamiento de Imágenes y Espectrogramas para Electrocardiogramas Fetales

Se realizó el procesamiento de imágenes de los electrocardiogramas para garantizar su adecuada visualización y análisis. Además, se llevaron a cabo pruebas con espectrogramas y procesamiento de audio, lo que permitió visualizar los datos de los electrocardiogramas de manera más detallada. Este proceso incluyó la conversión de señales de audio en representaciones visuales almacenadas en Google Drive a través de Google Colab, demostrando la correcta ejecución y éxito del procesamiento del espectrograma y el audio.

Durante este procedimiento, se lograron generar y visualizar las cuatro capas que representan los latidos de la madre, brindando información detallada y segmentada de las señales analizadas. Para mejorar la precisión del procesamiento, se emplearon principios matemáticos basados en la escala Mel. Esta escala es el resultado de una transformación no lineal de la escala de frecuencia y está diseñada de manera que los

sonidos equidistantes en la escala de Mel también "suenen" equidistantes para el oído humano.

La escala Mel permitió una mejor representación de las frecuencias relevantes para el análisis auditivo y visual de los electrocardiogramas. Como resultado, se lograron imágenes reescaladas en colores, las cuales fueron generadas exitosamente y almacenadas. Este proceso destaca la aplicación de métodos avanzados de procesamiento de señales y técnicas matemáticas para garantizar una representación precisa y comprensible de los datos obtenidos.

A continuación, se muestran las imágenes de los cuatro diferentes espectrogramas obtenidos con las señales, en los que se muestra el tiempo de la señal del electrocardiograma.

En la figura 4.20 se aprecia el primer espectrograma, en la cual se aprecia la primera sección de la clasificación del sonido del latido del corazón, donde lo marcado en color negro representa el silencio, el color naranja representa la onda sinusoidal y el tono morado representa el sonido de los latidos.

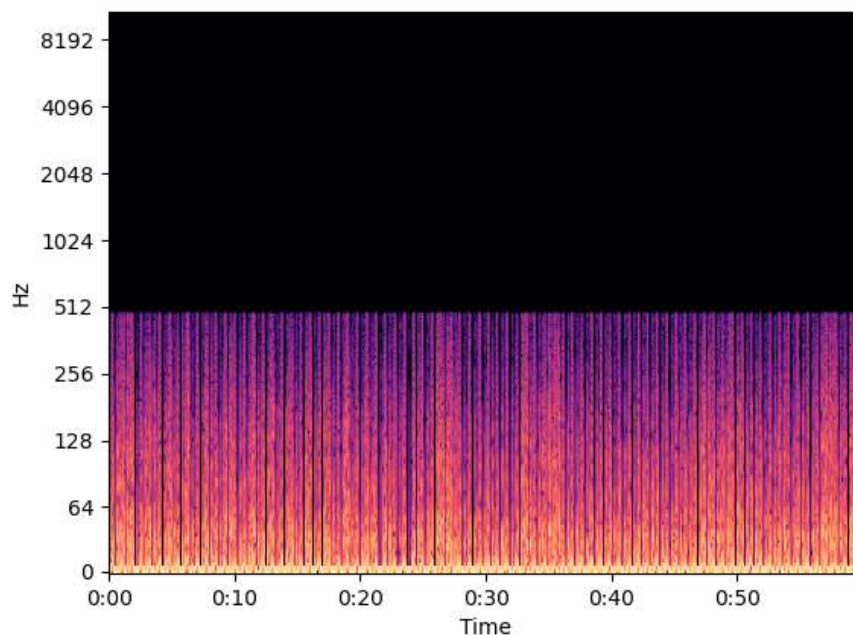


Figura 4.20 Espectrograma 1

La figura 4.21 detalla el segundo espectrograma el cual muestra la segunda señal de los latidos del corazón, se observan tiempo y sonido, con un ligero cambio.

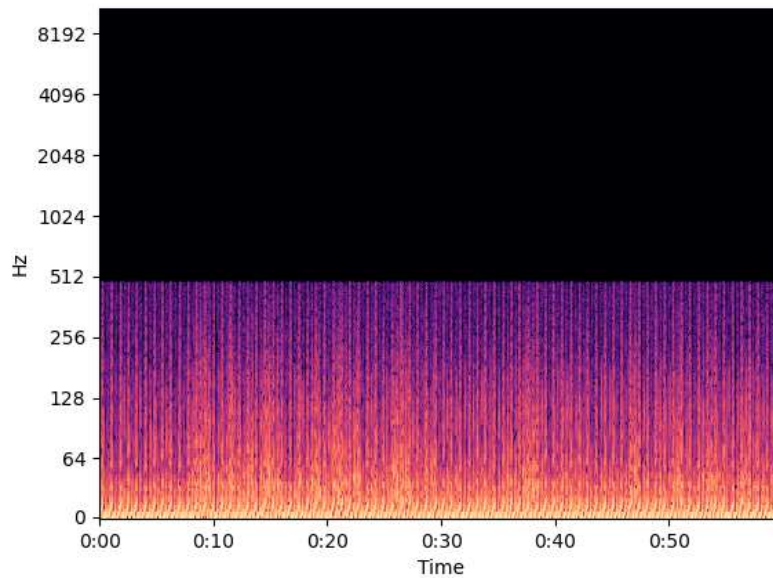


Figura 4.21 Espectrograma 2

En la figura 4.22 se puede ver el espectrograma 3, que es la tercera sección de las señales, se muestra el sonido y latidos del corazón, muestra un ligero cambio en los HZ del sonido.

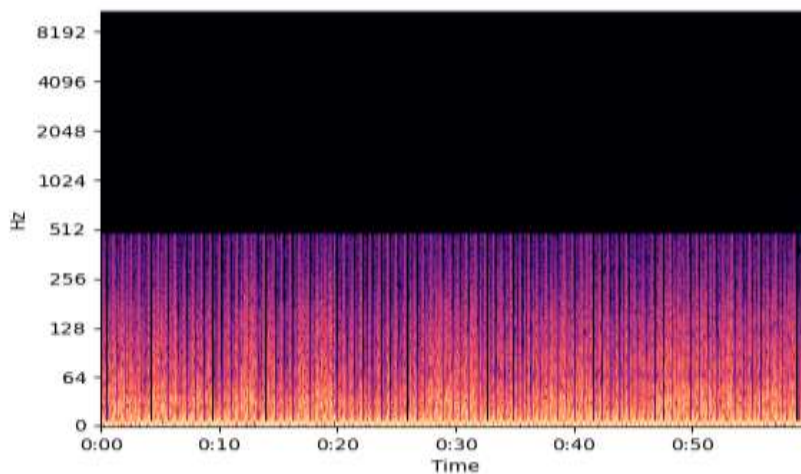


Figura 4.22 Espectrograma 3

En la figura 4.23 se observa la última señal de nuestra sección de señales con tiempo, sonido y latidos, con variaciones poco notables pero detectables, estas imágenes se procesaron con la red neuronal, la cual ayudo a convertir las imágenes para comprender los electrocardiogramas de los pacientes para los historiales médicos.

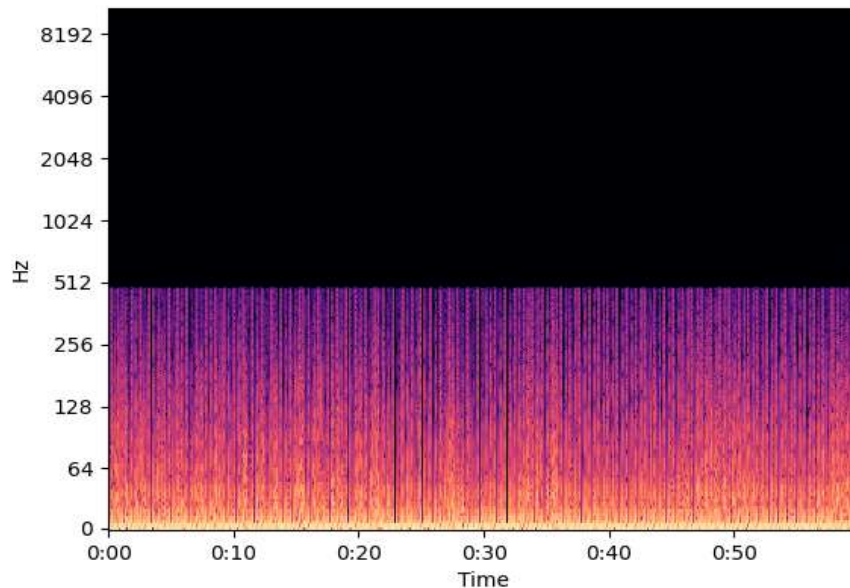


Figura 4.23 Espectrograma 4

4.6 Análisis de Riesgos

El desarrollo e implementación de un sistema web con integración de una red neuronal para el análisis de imágenes de electrocardiogramas representa una solución innovadora, pero también conlleva ciertos riesgos que deben identificarse y mitigarse para garantizar su éxito. A continuación, se presentan los principales riesgos asociados al proyecto y las estrategias de mitigación recomendadas.

1. Riesgos Técnicos

Integración entre la red neuronal y la página web

- **Riesgo:** la conexión entre la página web y la red neuronal puede presentar fallos, como tiempos de respuesta lentos, errores en la transmisión de datos o incompatibilidad entre tecnologías.
- **Impacto:** retrasos en el procesamiento de imágenes y resultados incorrectos para los médicos.
- **Mitigación:**
 - Realizar pruebas de integración utilizando APIs bien documentadas.
 - Optimizar el modelo de la red neuronal para reducir tiempos de procesamiento.

Errores en el Procesamiento de Imágenes

- **Riesgo:** el modelo de la red neuronal puede generar predicciones inexactas o erróneas debido a un entrenamiento insuficiente o datos de baja calidad.
- **Impacto:** posibles diagnósticos incorrectos que afectan la toma de decisiones médicas.
- **Mitigación:**
 - Entrenar el modelo con un conjunto de datos amplio y diverso que represente múltiples casos clínicos.
 - Actualizar periódicamente el modelo con nuevos datos y retroalimentación de los usuarios.

Fallas en la base de datos

- **Riesgo:** pérdida o corrupción de datos médicos debido a errores en la base de datos.
- **Impacto:** imposibilidad de acceder a los historiales clínicos, afectando la continuidad del servicio.
- **Mitigación:**
 - Implementar respaldos automáticos de la base de datos.
 - Utilizar un sistema de base de datos escalable y con disponibilidad.
 - Monitorear constantemente el estado de la base de datos para detectar anomalías.

Conclusiones y recomendaciones

El desarrollo del sistema de gestión de historiales médicos obstétricos ha permitido cumplir con los objetivos planteados en este proyecto. Se logró diseñar e implementar una plataforma eficiente para el almacenamiento y consulta de historiales clínicos de mujeres embarazadas.

En relación con gestionar información básica de pacientes y su historial clínico, el sistema permite un registro organizado y de fácil acceso, optimizando la atención médica al disponer de datos actualizados en tiempo real.

Se logró una interfaz que permite a los especialistas visualizar los resultados de manera clara y estructurada, lo que contribuye a una mejor toma de decisiones en el diagnóstico de posibles cardiopatías congénitas.

El sistema brinda herramientas que permiten correlacionar datos clínicos, favoreciendo un enfoque preventivo y mejorando la calidad de atención prenatal.

Finalmente, la implementación del backend con Python y Java, junto con una base de datos NoSQL en MongoDB, ha permitido un procesamiento rápido y eficiente de la información, asegurando la escalabilidad del sistema para futuras mejoras e integraciones.

Recomendaciones

1. **Ampliar la cobertura del sistema:** se recomienda extender la funcionalidad para incluir otros perfiles de pacientes y expandir su uso en diferentes instituciones de salud.
2. **Capacitación de los usuarios:** es importante diseñar programas de capacitación para los profesionales de la salud con el fin de maximizar el aprovechamiento del sistema y garantizar un uso adecuado de la información almacenada.
3. **Optimizar la seguridad y privacidad:** se recomienda fortalecer los protocolos de seguridad para garantizar la protección de los datos sensibles, implementando mejores prácticas en encriptación y acceso a la información.
4. **Análisis predictivo:** como una mejora futura, podría implementarse inteligencia artificial para identificar patrones en los datos clínicos y mejorar la detección temprana de enfermedades congénitas.

5. **Evaluaciones periódicas del sistema:** se sugiere realizar auditorías y pruebas continuas para evaluar el rendimiento del sistema y detectar posibles áreas de mejora en su funcionalidad y usabilidad.

Recomendaciones para la mejora del proyecto

Integración de la red neuronal con la plataforma web

Es fundamental conectar la red neuronal con la página web para automatizar el análisis de electrocardiogramas en tiempo real. Esto implicaría:

- Enviar imágenes desde la plataforma a la red neuronal mediante una API.
- Procesar las imágenes y obtener diagnósticos preliminares.
- Retornar los resultados analizados al historial clínico del paciente.

Notificaciones en Tiempo Real

Implementar un sistema de alertas automáticas para notificar a los médicos sobre anomalías detectadas en los electrocardiogramas, permitiendo una respuesta rápida ante posibles riesgos.

Optimización de la Experiencia del Usuario

Para mejorar la usabilidad del sistema, se recomienda:

- Diseñar una interfaz más intuitiva y adaptable a dispositivos móviles.
- Mejorar la navegación entre módulos para agilizar la gestión de historiales médicos.
- Automatizar la generación de reportes clínicos para reducir la carga de trabajo manual.

Seguridad y Privacidad de la Información

Dado que el sistema maneja datos médicos sensibles, se recomienda reforzar las medidas de seguridad mediante:

- Controles de acceso más estrictos para proteger la información de los pacientes.
- Implementación de cifrado en la base de datos y en las comunicaciones.

Escalabilidad del Sistema

Para garantizar el crecimiento del sistema a largo plazo, se recomienda:

- Migrar la base de datos a una infraestructura más robusta que permita manejar grandes volúmenes de información.
- Optimizar el rendimiento de las consultas para mejorar la velocidad de acceso a los historiales médicos.

En conclusión, este proyecto sienta las bases para un sistema de apoyo en el diagnóstico de patologías cardíacas fetales mediante el análisis automatizado de señales médicas. La implementación de mejoras futuras permitirá optimizar su funcionamiento y ampliar su impacto en el sector salud, beneficiando a médicos y pacientes mediante un acceso más eficiente y preciso a la información clínica.

Referencias

Córdova Ramírez, J. F. (2021). Propuesta de sistema de gestión documental de historiales médicos con aplicación de la firma digital. Caso: Hospital Daniel Alcides Carrión.https://alicia.concytec.gob.pe/vufind/Record/UNMS_5357f276e5bddfd93c88bd89456123d9

Azcona, L. (2009). El electrocardiograma. López Farré A, Macaya Miguel C, directores. Libro de la salud cardiovascular del Hospital Clínico San Carlos y la fundación BBVA. 1ª ed. Bilbao: Fundación BBVA, 49-56. https://www.fbbva.es/wp-content/uploads/2017/05/dat/DE_2009_salud_cardiovascular.pdf

De, G. (2019). Obstetricia y Ginecología.

Guzmán, F., & Arias, C. A. (2012). La historia clínica: elemento fundamental del acto médico. Revista colombiana de cirugía, 27(1), 15-24.

<https://www.revistacirugia.org/index.php/cirugia/article/view/195>

Kumar, V., Cotran, R. S., & Robbins, S. L. (2008). Patología humana. Elsevier Health Sciences.[https://scholar.google.com.mx/scholar?q=Kumar,+V.,+Cotran,+R.+S.,+%26+Robbins,+S.+L.+\(2008\).+Patolog%C3%ADa+humana.+Elsevier+Health+Sciences.&hl=es&as_sdt=0&as_vis=1&oi=scholar](https://scholar.google.com.mx/scholar?q=Kumar,+V.,+Cotran,+R.+S.,+%26+Robbins,+S.+L.+(2008).+Patolog%C3%ADa+humana.+Elsevier+Health+Sciences.&hl=es&as_sdt=0&as_vis=1&oi=scholar)

Pedroso Herrera, Tomás. “? ¿Qué es Internet?” Comunicar, no. 26, Jan. 2006, p. 226. GaleAcademicOneFile,link.gale.com/apps/doc/A197857048/AONE?u=anon~e7e0ae03&sid=googleScholar&xid=1f2da830.Accessed 30 Oct. 2024.<https://www.revista.unam.mx/vol.7/num6/art46/art46-1.htm>

Vidal Ledo, M. J., Rodríguez Dopico, R. M., & Martínez Hernández, G. (2014). Sistemas de gestión del aprendizaje. Educación Médica Superior, 28(3), 603-615. https://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21412014000300019

Otaola Barranquero, M. D. P., & Huete García, A. (2021). Riesgos de discriminación por discapacidad en la digitalización de la asistencia sanitaria. Teknokultura: Revista de Cultura Digital y Movimientos Sociales, 18(2), 115-125.

<https://revistas.ucm.es/index.php/TEKN/article/view/74401>

Ezeiza Pohl, C. E., Rampello, S. M., & Mecca, D. E. (2024). La transferencia del conocimiento en la universidad: aportes para su definición teórica y construcción de indicadores de ciencia y tecnología. <https://repositoriocyf.unlam.edu.ar/handle/123456789/1837>

Montávez Sánchez, M. (2024). Arquitectura de programación web: backend. <https://crea.ujaen.es/browse?type=author&value=Segura+S%C3%A1nchez%2C+Rafael>

Jiménez, J. A. A. Piensa en Haskell y en Python. Cocha Ger, M. I. (2024). Desarrollo de un Sistema para la Gestión de Delegaciones en la Unidad Nacional de Ciberdelitos: Frontend. <http://bibdigital.epn.edu.ec/handle/15000/26127>

Males Maldonado, P. E. (2024). Desarrollo de sistema para gestión de ventas de la microempresa "Mbyte Soluciones Tecnológicas": aplicación móvil. <http://bibdigital.epn.edu.ec/handle/15000/25575>

Congo Folleco, M. S. (2024). Desarrollo de una aplicación web para la renta de vehículos: frontend. <http://bibdigital.epn.edu.ec/handle/15000/25568>

Pérez Ibarra, S. G., Quispe, J. R., Mullicundo, F. F., & Lamas, D. A. (2021). Herramientas y tecnologías para el desarrollo web desde el FrontEnd al BackEnd. In XXIII Workshop de Investigadores en Ciencias de la Computación (WICC 2021, Chilecito, La Rioja). <http://sedici.unlp.edu.ar/handle/10915/120476>

Jindal, A., Emani, K. V., Daum, M., Poppe, O., Haynes, B., Pavlenko, A., ... & Patel, H. (2021, February). Magpie: Python at Speed and Scale using Cloud Backends. In CIDR. https://www.cidrdb.org/cidr2021/papers/cidr2021_paper08.pdf

Caballero, J. J. V. (2016). Modelo de procesos para el desarrollo del front-end de aplicacionesweb. *Interfases*,(9),187-208. <https://www.redalyc.org/pdf/7301/730180368008.pdf>

Pérez,JE(2019).IntroduccióndeJavaScript. https://dspace.itsjapon.edu.ec/jspui/bitstream/123456789/430/1/introduccion_javascript%20%281%29.pdf

Glosario de Términos.

Electrocardiograma (ECG): Registro gráfico de la actividad eléctrica del corazón utilizado para detectar anomalías en su funcionamiento.

Ecocardiograma: Prueba diagnóstica que utiliza ultrasonido para evaluar la estructura y función del corazón.

Historial Médico: Registro detallado de la información clínica de un paciente, incluyendo antecedentes, diagnósticos y tratamientos.

Red Neuronal Convolucional (CNN): Tipo de inteligencia artificial utilizada para el procesamiento de imágenes, como electrocardiogramas.

Procesamiento de Señales: Técnica empleada para analizar y transformar señales médicas en datos útiles.

Espectrograma de Mel: Representación visual de una señal de audio que resalta características clave para su análisis.

Interfaz de Usuario (UI): Elemento visual con el que interactúan los usuarios en una plataforma digital.

Base de Datos NoSQL: Sistema de almacenamiento de datos que permite manejar información de manera flexible y escalable.

Hosting: Servicio que permite alojar páginas web en servidores accesibles en línea.

Seguridad de Datos: Conjunto de medidas para proteger la información sensible de accesos no autorizados.

Glosario de Acrónimos

API (Application Programming Interface): Interfaz de programación que permite la comunicación entre sistemas.

CSV (Comma-Separated Values): Formato de archivo para almacenar datos tabulares en texto plano.

ECG (Electrocardiograma): Registro de la actividad eléctrica del corazón.

CNN (Convolutional Neural Network): Red neuronal convolucional usada para análisis de imágenes.

UI (User Interface): Interfaz de usuario en una aplicación web o sistema digital.

NoSQL (Not Only SQL): Tipo de base de datos flexible y escalable.

AI (Artificial Intelligence): Inteligencia artificial utilizada para procesar información.

ML (Machine Learning): Aprendizaje automático basado en modelos matemáticos y estadísticos.

DBMS (Database Management System): Sistema de gestión de bases de datos.

SSL (Secure Sockets Layer): Protocolo de seguridad para cifrar la comunicación en internet.

HTTP (Hypertext Transfer Protocol): Protocolo de transferencia de datos en la web.

HTTPS (Hypertext Transfer Protocol Secure): Versión segura de HTTP con cifrado SSL/TLS.

REST (Representational State Transfer): Arquitectura de software usada en APIs.

JSON (JavaScript Object Notation): Formato ligero de intercambio de datos.

XML (Extensible Markup Language): Lenguaje de marcado utilizado para almacenar y transportar datos.

FTP (File Transfer Protocol): Protocolo para la transferencia de archivos entre sistemas.

TCP/IP (Transmission Control Protocol / Internet Protocol): Conjunto de protocolos de comunicación en redes.

DB (Database): Base de datos utilizada para almacenar información estructurada.

CRUD (Create, Read, Update, Delete): Operaciones básicas en bases de datos.

JWT (JSON Web Token): Mecanismo de autenticación basado en tokens.

AES (Advanced Encryption Standard): Algoritmo de cifrado seguro para datos sensibles.

OCR (Optical Character Recognition): Tecnología para el reconocimiento de texto en imágenes.

GUI (Graphical User Interface): Interfaz gráfica de usuario en sistemas digitales.

UX (User Experience): Experiencia del usuario al interactuar con un sistema.

AWS (Amazon Web Services): Plataforma de servicios en la nube.

VM (Virtual Machine): Máquina virtual utilizada para emulación de sistemas.

PaaS (Platform as a Service): Modelo de computación en la nube que ofrece plataformas de desarrollo.

IaaS (Infrastructure as a Service): Infraestructura como servicio en la nube.

SaaS (Software as a Service): Software como servicio accesible desde internet.

VPN (Virtual Private Network): Red privada virtual que protege la conexión en internet.