UNIVERSIDAD PRIVADA DE SANTA CRUZ DE LA SIERRA

FACULTAD DE INGENIERÍA

Carrera: Ingeniería Electrónica

TRABAJO FINAL DE GRADO

Modalidad: Proyecto de Grado

DISEÑO E IMPLEMENTACIÓN DE UN ASISTENTE PARA DIAGNÓSTICO MÉDICO USANDO INTELIGENCIA ARTIFICIAL PARA LAS SALAS DE EMERGENCIA DE LOS HOSPITALES PÚBLICOS

Andres Ignacio Justiniano Ortega

Santa Cruz de la Sierra – Bolivia

2025

UNIVERSIDAD PRIVADA DE SANTA CRUZ DE LA SIERRA

FACULTAD DE INGENIERÍA

CARRERA: INGENIERÍA ELECTRÓNICA

TRABAJO FINAL DE GRADO

Modalidad: Proyecto de Grado

DISEÑO E IMPLEMENTACIÓN DE UN ASISTENTE PARA DIAGNÓSTICO MÉDICO USANDO INTELIGENCIA ARTIFICIAL PARA LAS SALAS DE EMERGENCIA DE LOS HOSPITALES PÚBLICOS

Andres Ignacio Justiniano Ortega

2021111636

Proyecto para optar por el grado de Licenciado en Ingeniería Electrónica

Santa Cruz de la Sierra – Bolivia

2025

AGRADECIMIENTOS

Primeramente, a Dios por ser mi guía y darme fortaleza para culminar este camino.

A mis padres, Sheila y Roberto, por su continuo e incondicional apoyo. Por inculcarme los valores y principios que me forman no solo como profesional sino también como persona. Por su sacrificio y por ser excelentes ejemplos de vida y perseverancia. Los amo y les agradezco con todo mi corazón.

A mis abuelos, Tato, Nena, Carola e Iván, por todo el cariño y por siempre animarme a seguir este camino. Era un sueño para ellos verme en esta instancia y aunque no puedan estar aquí, este logro es tanto mío como de ellos. En mi memoria siempre estarán presentes.

A mis compañeros de universidad por la compañía, por las horas compartidas y por hacer que todo el trayecto de mis estudios este lleno de risas, conversaciones interminables y bonitos recuerdos.

Al laboratorio de electrónica por ser el lugar donde me desarrolle como profesional y por ser el lugar donde están la mayoría de mis recuerdos de mi vida universitaria.

A la Universidad Privada de Santa Cruz de la Sierra (UPSA) por toda la educación brindada y por formarme como profesional.

A la mujer que siempre me acompaña y que siempre está para mí, a mi novia Andrea. Tu presencia me inspira a ser mejor cada día y tu amor es todo lo que necesito para salir adelante. Gracias por siempre ser vos.

ABSTRACT

Titulo	Diseño e implementación de un asistente para diagnóstico médico usando inteligencia artificial para las salas de emergencia de los hospitales públicos.
Autor	Andres Ignacio Justiniano Ortega

PROBLEMÁTICA

Sobrecarga del personal médico debido a la alta demanda de atención, la falta de recursos y problemas en la administración del sistema de salud, lo que genera largos tiempos de espera y afecta la calidad de la atención.

OBJETIVO GENERAL

Implementar un asistente para diagnóstico médico usando inteligencia artificial para optimizar la atención en las salas de emergencia de los hospitales públicos de Bolivia.

CONTENIDO

El trabajo de grado se divide en diez capítulos, teniendo en cuenta lo planteado en los objetivos específicos.

Capítulo 1: Introducción

Capítulo 2: Salas de emergencia en Bolivia

Capítulo 3: Modelos de inteligencia artificial

Capítulo 4: Diseño del asistente para diagnóstico médico

Capítulo 5: Implementación del asistente para diagnóstico médico

Capítulo 6: Pruebas y análisis de resultados

Capítulo 7: Conclusiones y recomendaciones

Capítulo 8: Impacto social y económico

Capítulo 9: Anexos

Capítulo 10: Referencias bibliográficas

CARRERA	Ingeniería Electrónica
DESCRIPTORES O TEMAS	Robótica, Inteligencia Artificial
E-MAIL	andresjustinianoortega@gmail.com
FECHA	Abril 2026

ÍNDICE

1. INTRODUCCIÓN.....2

1.1. Introducción.....2

1.2. Situación problemática.....2

1.3. Situación deseada.....3

1.4. Mapa mental.....4

1.5. Formulación problemática.....4

1.6. Objetivos.....5

1.6.1. Objetivo general.....5

1.6.2. Objetivos específicos.....5

1.7. Metodología.....5

2. SALAS DE EMERGENCIA EN BOLIVIA.....8

2.1. Definición de las salas de emergencia.....8

2.2. Función de las salas de emergencia.....8

2.3. Situación actual de las salas de emergencia en Bolivia.....9

2.3.1. Enfermedades y síntomas más frecuentes en emergencias.....10

2.3.2. Estadísticas y datos relevantes.....11

2.4. Tecnologías en las salas de emergencia.....12

2.5. Retos actuales.....14

3. MODELOS DE INTELIGENCIA ARTIFICIAL.....16

3.1. Caracterización de la inteligencia artificial.....16

3.1.1. Definición.....16

3.1.2. Funcionamiento 16
3.1.3. Tipos de inteligencia artificial 17
3.2. Modelos aplicables de inteligencia artificial 17
3.2.1. Sistemas basados en reglas 17
3.2.2. Redes neuronales artificiales 18
3.2.3. Algoritmos de aprendizaje automático (ML) 18
3.2.3.1. Aprendizaje Supervisado 18
3.2.3.2. Aprendizaje no supervisado 18
3.2.3.3. Aprendizaje por esfuerzo 19
3.2.4. Modelo de aprendizaje profundo 19
3.2.5. Modelo de NLP 19
3.3. Inteligencia artificial en la medicina 19
3.3.1. Impacto en el área de la medicina 20
3.3.2. Modelos aplicados al diagnóstico médico 20
3.3.3. Consideraciones éticas y legales de la inteligencia artificial en la medicina 21
4. DISEÑO DEL ASISTENTE PARA DIAGNÓSTICO MÉDICO 24
4.1. Descripción del proceso 24
4.2. Instrumentación requerida 24
4.2.1. Raspberry Pi 4 25
4.2.1.1. Sistema operativo 27
4.2.1.2. Programación 27

4.2.2. Pantalla táctil HMTECH 10.1 27
4.2.3. Sensor de temperatura corporal MLX90614 28
4.2.4. Sensor para latidos del corazón COMIMARK 29
4.2.4.1. Filtro paso bajo 30
4.2.4.2. Convertidor ADC MCP3008 30
4.3. Análisis de conexiones 32
4.4. Diseño del asistente 33
4.4.1. Hardware 33
4.4.1.1. Conexiones del sistema 33
4.4.1.2. Diseño de circuitos del sistema 42
4.4.1.3. Diseño 3D del asistente 46
4.4.2. Software 48
4.4.2.1. Diseño de la interfaz gráfica 48
4.4.2.2. Requerimientos de la interfaz 49
4.4.2.3. Desarrollo de la interfaz 50
4.4.3. Sistema en conjunto 52
5. IMPLEMENTACIÓN DEL ASISTENTE PARA DIAGNÓSTICO MÉDICO 55
5.1. Montaje del prototipo 55
5.1.1. Fabricación de la placa electrónica 55
5.1.2. Conexión de los componentes 58
5.1.3. Verificación de funcionamiento 59

5.2. Conectividad y almacenamiento 59
5.2.1. Configuración para comunicación entre interfaces 59
5.2.2. Integración con inteligencia artificial 60
5.2.3. Implementación de base de datos 60
5.3. Desarrollo del software 61
5.3.1. Estructura general del sistema 61
5.3.2. Desarrollo del software 61
5.3.2.1. Lectura del sensor de temperatura 62
5.3.2.2. Lectura del sensor de pulso 62
5.3.2.3. Conexión con la inteligencia artificial 64
5.3.3. Verificación de funcionamiento 65
5.4. Ensamblaje del prototipo 69
5.4.1. Impresión de la estructura del prototipo 69
5.4.2. Armado del asistente 71
5.5. Costo de implementación 71
6. PRUEBAS Y ANÁLISIS DE RESULTADOS 74
6.1. Pruebas funcionales de hardware 74
6.1.1. Validación del circuito electrónico 74
6.1.2. Validación del sensor de pulso 75
6.1.3. Validación del sensor de temperatura 76
6.2. Pruebas de integración y rendimiento del sistema 77
6.2.1. Prueba de rendimiento de software 77

6.2.2. Prueba de interacción con la interfaz gráfica 78
6.3. Prueba de precisión del módulo de diagnóstico 79
6.3.1. Elaboración de los casos de prueba 79
6.3.2. Ejecución de diagnóstico 80
6.3.3. Comparación de diagnóstico 81
6.4. Evaluación de resultados 82
6.4.1. Análisis del hardware 82
6.4.2. Análisis del software y rendimiento 83
6.4.3. Análisis de la precisión diagnóstico 83
7. CONCLUSIONES Y RECOMENDACIONES 86
7.1. Conclusiones 86
7.2. Recomendaciones 87
8. IMPACTO SOCIAL Y ECONÓMICO 90
9. ANEXOS 92
9.1. Código completo del asistente en Python 92
9.2. Hoja de datos del sensor MLX90614 92
9.3. Hoja de datos del convertidor ADC MCP3008 93
9.4. Hoja de datos del amplificador operacional LM358 94
10. REFERENCIAS BIBLIOGRÁFICAS 96
CURRÍCULUM VITAE 102

ÍNDICE DE ILUSTRACIONES

Ilustración 1.1. Mapa mental....4
Ilustración 4.1. Raspberry Pi Modelo B....25
Ilustración 4.2. Diagrama de conexiones de la Raspberry Pi 4....26
Ilustración 4.3. Pines GPIO de la Raspberry Pi 4 Modelo B....26
Ilustración 4.4. Pantalla táctil HMTECH 10.1....28
Ilustración 4.5. Sensor para medir temperatura corporal MLX90614....28
Ilustración 4.6. Pulse Heart Rate Sensor COMIMARK....29
Ilustración 4.7. Filtro paso bajo....30
Ilustración 4.8. Convertidor ADC MCP3008....31
Ilustración 4.9. Pines del convertidor ADC MCP3008....31
Ilustración 4.10. Diagrama de conexiones....32
Ilustración 4.11. Conectores físicos de la pantalla HMETECH....33
Ilustración 4.12. Conexión entre la Raspberry Pi y la pantalla HMTECH 34
Ilustración 4.13. Pines para conectar el sensor MLX90614....34
Ilustración 4.14. Pines para medición de pulso COMIMARK....35
Ilustración 4.15. Arquitectura tipo Sallen-Key de segundo orden....37
Ilustración 4.16. Componente LM358....39
Ilustración 4.17. Pines del LM358....39
Ilustración 4.18. Diseño del filtro completo....40
Ilustración 4.19. Conexiones del convertidor ADC MCP3008....41
Ilustración 4.20. Diseño del circuito del sistema....43
Ilustración 4.21. Conexión de los pines GPIO en la Raspberry Pi....43
Ilustración 4.22. Diseño de PCB....44
Ilustración 4.23. Diseño 3D de la placa electrónica....45
Ilustración 4.24. Trazas de la placa electrónica....45

Ilustración 4.25. Estructura del asistente 46
Ilustración 4.26. Diseño 3D de la pantalla 47
Ilustración 4.27. Diseño 3D del sensor de temperatura 47
Ilustración 4.28. Diseño 3D del sensor de pulso 47
Ilustración 4.29. Parte trasera del asistente 48
Ilustración 4.30. Boceto para la adquisición de síntomas 50
Ilustración 4.31. Boceto para la adquisición de datos extra del paciente .. 51
Ilustración 4.32. Boceto para la medición de temperatura y pulso cardiaco 51
Ilustración 4.33. Boceto donde se muestra el resultado del análisis 52
Ilustración 4.34. Simulación de los componentes del sistema 53
Ilustración 5.1. Circuito en Protoboard 55
Ilustración 5.2. Trazas sobre la placa de cobre 56
Ilustración 5.3. Placa perforada 56
Ilustración 5.4. Componentes en la placa 57
Ilustración 5.5. Soldadura de los componentes 57
Ilustración 5.6. Conexión de los sensores 58
Ilustración 5.7. Conexión de pantalla táctil con Raspberry Pi 58
Ilustración 5.8. Verificación de funcionamiento de conexiones 59
Ilustración 5.9. Código para lectura del sensor MLX90614 62
Ilustración 5.10. Algoritmo de BPM para el sensor de pulso 63
Ilustración 5.11. Implementación de IA en la programación 64
Ilustración 5.12. Instrucciones de redacción para Gemini 64
Ilustración 5.13. Diseño del menú principal 65
Ilustración 5.14. Pestaña para añadir síntomas 66
Ilustración 5.15. Pestaña de antecedentes médicos 66
Ilustración 5.16. Pestaña para medición de signos vitales 67

Ilustración 5.17. Pestaña de espera 67
Ilustración 5.18. Pestaña de resultados..... 68
Ilustración 5.19. Pestaña de historial de pacientes 69
Ilustración 5.20. Modelado en 3D completo..... 70
Ilustración 5.21. Estructura impresa en 3D..... 70
Ilustración 5.22. Asistente terminado y ensamblado 71
Ilustración 6.1. Análisis del circuito con generador de señales 74
Ilustración 6.2. Análisis del circuito con el sensor conectado 75
Ilustración 6.3. Mensaje de retroalimentación de la inteligencia artificial 77

ÍNDICE DE TABLAS

Tabla 2.1. Donde acude la población cuando tiene problemas de salud 11
Tabla 2.2. Protocolo de Adecuación de Emergencias y/o Urgencias Hospitalarias.....13
Tabla 4.1. Módulos de software del asistente. 49
Tabla 5.1. Tabla de librerías para el sensor MLX90614 62
Tabla 5.2. Tabla de librerías para el sensor de pulso 63
Tabla 5.3. Tabla de costos de materiales 72
Tabla 5.4. Costos de mano de obra del prototipo 72
Tabla 6.1. Pruebas del sensor de pulso.....76
Tabla 6.2. Pruebas del sensor de temperatura.....76
Tabla 6.3. Tiempo de espera de respuesta del sistema y conexión con IA .78
Tabla 6.4. Prueba de interacción del asistente 78
Tabla 6.5. Casos de prueba 79
Tabla 6.6. Diagnóstico del asistente.....80
Tabla 6.7. Comparación del nivel de emergencia de los casos de prueba..81
Tabla 6.8. Comparación del diagnóstico de los casos de prueba.....82

RESUMEN EJECUTIVO

El presente trabajo tiene como objetivo el desarrollo de un asistente para las salas de emergencias para optimizar la atención en los hospitales públicos de Bolivia. Mediante inteligencia artificial realiza diagnósticos médicos en base a información que proveen los pacientes. La motivación principal del proyecto surge debido a la fragilidad del sistema de salud actual. Largas filas, alta demanda de atención y la escasez de personal generan inconvenientes en el tratamiento de los pacientes.

El sistema propuesto hace uso de una Raspberry Pi 4 modelo B para integrar varios dispositivos electrónicos para formar el asistente. El paciente detalla su sintomatología ante una pantalla que registra toda la información relevante. Se usa un sensor de temperatura MLX90614 y un sensor de pulso COMIMARK para brindar datos biométricos al asistente y tener un mejor análisis. Se implementa un filtro de segundo orden Sallen-key para mejorar la señal del sensor de pulso y un convertidor MCP3008 para llevar de analógica a digital la información. El diseño se realiza en 3D y se fabricó mediante impresión en filamento ABS. El circuito electrónico se desarrolló en una placa de cobre diseñada en el software Proteus y fabricada mediante mecanizado CNC.

Las pruebas de validación se realizaron mediante comparaciones con dispositivos comerciales y con personal médico con experiencia en el área, alcanzando una tasa de precisión del 90 %.

El asistente es una herramienta viable para optimizar y apoyar los procesos llevados a cabo en el área de emergencias. Ayuda a reducir la carga del personal médico y mejora los tiempos de atención en el contexto nacional.

CAPÍTULO 1

INTRODUCCIÓN

1. INTRODUCCIÓN

1.1. Introducción

La inteligencia artificial es un elemento fundamental en la atención médica moderna y se utiliza para respaldar a los profesionales de la salud en entornos clínicos y en investigaciones en curso. En la actualidad, una de las funciones más comunes de la inteligencia artificial en el ámbito médico es el apoyo a la toma de decisiones clínicas.

En Bolivia, se están realizando esfuerzos significativos para fortalecer el sistema de información en salud y avanzar en la transformación digital del sector sanitario. No obstante, diversos obstáculos de índole social y política han ralentizado el progreso esperado en los hospitales.

En este contexto, la implementación de un asistente para diagnóstico médico representa una alternativa prometedora para optimizar los procesos hospitalarios. En los hospitales públicos, las salas de emergencias suelen experimentar una alta saturación, lo que genera desequilibrios en la atención y malestar en la población, que demanda soluciones rápidas a sus afecciones.

Este prototipo tiene como finalidad colaborar en la gestión de datos y la administración de pacientes, con el objetivo de mejorar la organización y la eficiencia en la atención sanitaria, sin sustituir al personal médico.

1.2. Situación problemática

En las salas de emergencia de los hospitales en Bolivia, uno de los principales desafíos es la sobrecarga del personal médico debido a la alta demanda de atención, lo que genera largos tiempos de espera para los pacientes. Esta problemática se ve agravada por la falta de recursos tecnológicos que

permitan optimizar los procesos iniciales de evaluación y clasificación de los síntomas.

Asimismo, el sistema de salud en Bolivia presenta deficiencias en su administración, lo que con frecuencia provoca situaciones en las que el personal sanitario se ve obligado a suspender sus actividades debido a retrasos en los pagos o a la escasez de suministros. Como consecuencia, los pacientes deben esperar largos períodos antes de ser atendidos, lo que puede comprometer la eficacia de la atención médica, especialmente en casos de urgencia.

En algunos hospitales, se presentan situaciones en las que un único médico general debe atender a 50 pacientes, con el apoyo de solo dos enfermeras para organizar y asistir a los recién llegados. Ante este panorama, resulta fundamental explorar soluciones tecnológicas innovadoras, como la implementación de un asistente para diagnóstico médico que permita a los pacientes ingresar sus síntomas y recibir recomendaciones preliminares. Esta herramienta contribuiría a reducir la carga del personal médico y a agilizar el proceso de atención en las salas de emergencia.

1.3. Situación deseada

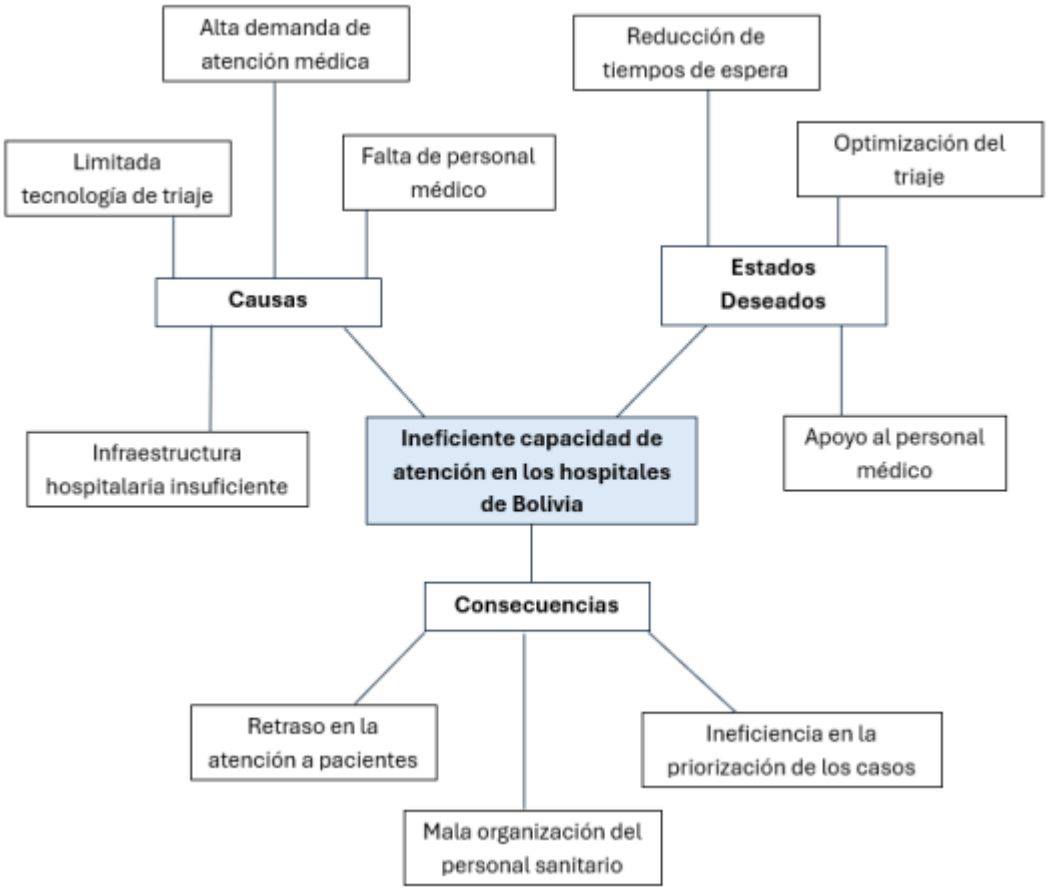
Reducir los tiempos de espera que se generan en los hospitales públicos a lo largo del país, y al mismo tiempo, optimizar el triaje en los centros de salud para que las personas puedan ser atendidas de forma más rápida y eficiente.

Dar apoyo al personal médico de los hospitales para así no sobrecargar tanto sus labores, además de determinar qué casos son de mayor urgencia que otros y que casos no requieren tanta atención.

1.4. Mapa mental

En la Ilustración 1.1. se muestra el mapa mental partiendo de la situación problemática:

Ilustración 1.1. Mapa mental



1.5. Formulación problemática

Implementación de un asistente para diagnóstico médico usando inteligencia artificial para agilizar y mejorar la atención en salas de emergencia de los hospitales públicos de Bolivia.

1.6. Objetivos

1.6.1. Objetivo general

Implementar un asistente para diagnóstico médico usando inteligencia artificial para optimizar la atención en las salas de emergencia de los hospitales públicos de Bolivia.

1.6.2. Objetivos específicos

- Caracterizar la situación actual de las salas de emergencia en los hospitales en Bolivia.
- Caracterizar y seleccionar el modelo de inteligencia artificial a utilizar.
- Diseñar el asistente para diagnóstico médico basado en inteligencia artificial para las salas de emergencia en los hospitales.
- Implementar el asistente para diagnóstico médico basado en inteligencia artificial para las salas de emergencia en los hospitales.

1.7. Metodología

-Caracterización de las salas de emergencia de Bolivia

- ❖ Entrevistas a profesionales de la salud y administradores de hospitales.
- ❖ Encuestas a personal médico y pacientes.
- ❖ Revisión de informes oficiales sobre atención en emergencias.
- ❖ Investigación sobre los protocolos y métodos de atención.

-Evaluación y selección del modelo de inteligencia artificial

- ❖ Investigación sobre modelos de IA.
- ❖ Revisión de la IA en la medicina
- ❖ Selección de modelos preentrenados.
- ❖ Pruebas de precisión y confiabilidad.

-Diseño del asistente para diagnóstico médico

- Evaluación y selección de la instrumentación necesaria
- Modelado y simulación del asistente en software de diseño.
- Desarrollo del sistema de inteligencia artificial para el análisis de síntomas ingresados por el paciente.
- Integración del software con bases de datos médicas y sistemas hospitalarios.

-Implementación del asistente para diagnóstico médico

- Instalación del asistente en un entorno relacionado.
- Capacitación del personal médico en el uso del asistente.
- Evaluación de la interacción del paciente con el asistente mediante observaciones y encuestas.

-Pruebas y análisis de resultados

- Pruebas de validación del asistente en escenarios de emergencia.
- Análisis de efectividad del sistema de IA en la interpretación de síntomas ingresados por los pacientes.

-Conclusiones y recomendaciones

- Evaluación del funcionamiento del asistente.
- Elaboración de recomendaciones para futuras mejoras y escalabilidad del sistema.

CAPÍTULO 2

SALAS DE EMERGENCIA EN BOLIVIA

2. SALAS DE EMERGENCIA EN BOLIVIA

2.1. Definición de las salas de emergencia

La emergencia según la OMS es aquel caso en que la falta de asistencia conduciría a la muerte en minutos y en el que la aplicación de primeros auxilios por cualquier persona es de importancia vital [1].

Se denomina sala de emergencia a las unidades médicas especializadas en la atención inmediata de pacientes que presentan condiciones críticas o situaciones de urgencia.

En ellas se pueden tratar traumatismos, tomar radiografías, realizar procedimientos quirúrgicos y tratar otras situaciones que ponen en riesgo la vida. La mayoría de los hospitales tienen una sala de emergencias que está abierta las 24 horas del día, los siete días de la semana [2].

2.2. Función de las salas de emergencia

El departamento de emergencia tiene como función la recepción, examen, valoración y tratamiento de pacientes que requieren atención médica de urgencia como resultado de un accidente o enfermedad repentina y que no pueden ser atendidos en consulta externa o ser internados en forma programada [3]. Su objetivo es estabilizar al paciente, diagnosticar su problema y decidir el siguiente paso en su tratamiento, ya sea hospitalización, cirugía o alta médica con seguimiento.

Se debe considerar un programa coordinado para cubrir cada paso necesario en el cuidado de los usuarios con serias heridas o enfermedades, desde su admisión, tratamiento, derivación dentro del hospital o referencia a otro nivel hasta su rehabilitación [1].

2.3. Situación actual de las salas de emergencia en Bolivia

El sistema de salud en Bolivia enfrenta una crisis profunda que afecta la calidad de los servicios médicos y pone en riesgo la atención de los pacientes. Uno de los principales problemas es la escasez de personal médico, lo que ha llevado al cierre de servicios esenciales en hospitales de referencia. Por ejemplo, en Santa Cruz, la sala de urgencias del Hospital Japonés fue cerrada debido a la falta de profesionales de salud, afectando a numerosos pacientes que dependen de este centro médico [4]. Esta situación también se refleja en otros hospitales, como el Oncológico, donde la atención se encuentra saturada y las condiciones son críticas debido a la insuficiencia de recursos y de personal [5].

Además, las deficiencias estructurales del sistema de salud han sido corroboradas por instituciones como la Autoridad de Supervisión de la Seguridad Social de Corto Plazo (ASUSS) y la Defensoría del Pueblo. Estas entidades han identificado problemas graves en los servicios de emergencia de los hospitales obreros de la Caja Nacional de Salud (CNS), donde los pacientes deben enfrentar largas esperas y condiciones inadecuadas para su atención [6].

Otro aspecto alarmante de la crisis es el retraso en las consultas y cirugías, lo que pone en peligro la vida de los pacientes. Según un informe de la Fundación Milenio, los tiempos de espera para recibir atención médica pueden extenderse hasta un mes, lo que agrava el estado de salud de quienes requieren atención inmediata [7]. Estos retrasos evidencian la sobrecarga del sistema y la falta de una planificación adecuada para responder a la demanda creciente de servicios de salud. Desde una perspectiva académica, investigaciones [8] han señalado que las fallas en el sistema de salud

boliviano se deben a una combinación de problemas financieros, administrativos y de gestión de recursos humanos.

2.3.1. Enfermedades y síntomas más frecuentes en emergencias

A lo largo de los años, las causas más frecuentes de asistencia a las salas de emergencia han experimentado variaciones, las principales afecciones atendidas en los hospitales suelen ser infecciones respiratorias, los traumatismos, las lesiones en la cabeza, las enfermedades gastrointestinales de origen infeccioso, las diarreas, las infecciones de las vías urinarias y la diabetes. En los hospitales del departamento de Santa Cruz se ha registrado un incremento en los casos de dengue. Según el último informe del Ministerio de Salud, se han reportado 6.453 casos confirmados de esta enfermedad y 18 fallecimientos a nivel nacional [9]. El dengue puede provocar fiebre alta, cefalea, vómitos y erupciones cutáneas, y en su forma hemorrágica representa un riesgo potencialmente mortal.

Si bien muchas de estas enfermedades continúan siendo recurrentes en los servicios de emergencia, en los últimos años se han sumado nuevas patologías a la lista. Entre ellas, se destacan diversas afecciones oculares y respiratorias originadas por la exposición al humo y la ceniza generados por los incendios forestales [10]. Asimismo, se han impulsado estrategias para promover prácticas seguras en el consumo de agua y alimentos, con el objetivo de prevenir infecciones gastrointestinales y enfermedades diarreicas en la población.

Además, resulta pertinente considerar el impacto que tuvo la pandemia de COVID-19 en la saturación y colapso del sistema de salud, así como la incidencia de brotes recientes de influenza, los cuales también han afectado a la población.

2.3.2. Estadísticas y datos relevantes

En la Tabla 2.1. se muestra la cantidad de personas que acuden a un centro de salud público cuando presentan algún malestar.

Tabla 2.1. Donde acude la población cuando tiene problemas de salud

DEPARTAMENTO, PROVINCIA Y MUNICIPIO	POBLACIÓN	CAJA DE SALUD (CNS, COSSMIL, OTRAS)
BOLIVIA	10.059.856	20,2
CHUQUISACA	**581.347**	**18,4**
Sucre	261.201	28,6
LA PAZ	**2.719.344**	**21,2**
La Paz	766.468	34,6
El Alto	848.452	22,3
COCHABAMBA	**1.762.761**	**17,6**
Cochabamba	632.013	24,1
ORURO	**494.587**	**22,4**
Oruro	264.943	26,9
POTOSÌ	**828.093**	**22,8**
Potosí	191.302	35,3
TARIJA	**483.518**	**24,3**
Tarija	205.375	30,8
SANTA CRUZ	**2.657.762**	**19,0**
Santa Cruz de la Sierra	1.454.539	22,5
BENI	**422.008**	**23,1**
Trinidad	106.596	27,5
PANDO	**110.436**	**17,4**
Cobija	46.267	29,3

NOTA: En la tabla se muestra el porcentaje de la población que acude a centros de salud públicos [11]

De acuerdo con el Ministerio de Salud y Deportes, la mayoría de los hospitales de tercer nivel en el país atienden un promedio de 150 a 300 pacientes por día en sus unidades de emergencia, lo que genera una alta carga asistencial, especialmente en zonas urbanas como La Paz, Cochabamba y Santa Cruz [12]. Esta saturación se ve agravada por el limitado número de camas, personal médico insuficiente y demoras en la atención, especialmente en los horarios nocturnos y fines de semana.

2.4. Tecnologías en las salas de emergencia

El avance tecnológico en el ámbito de la medicina ha generado beneficios significativos tanto para los pacientes como para el personal de salud. En Bolivia, la adopción e implementación de tecnología médica posee un alto potencial transformador para el sector salud. No obstante, su desarrollo está condicionado a la formulación e implementación de una estrategia integral y coordinada en distintos niveles gubernamentales [13].

Se han implementado otros avances como el Programa de Telesalud, como una forma rápida y eficaz de intercambio de información médica, con servicios de tele consultas, tele diagnósticos y capacitación continua de un punto a otro. El objetivo del Programa de Telesalud es contribuir a la reducción de la exclusión social promoviendo alternativas adecuadas para mejorar el acceso a la salud, superando las barreras geográficas y la optimización de las destrezas y capacidades de profesionales especializados y subespecialidades en el marco de la política de Salud SAFCI [14].

En el panorama actual, la tecnología médica emerge como una herramienta vital para elevar la calidad y accesibilidad de la atención sanitaria a nivel global. Bolivia enfrenta retos específicos para su efectiva implementación, incluyendo limitaciones en infraestructura tecnológica, restricciones financieras y la necesidad de capacitación del personal sanitario además de las disparidades geográficas y socioeconómicas [13].

Uno de los protocolos comúnmente utilizados en el país es el Protocolo de Adecuación de Emergencias y/o Urgencias Hospitalarias (PAUH), está compuesto por cinco apartados que incluyen criterios de gravedad, tratamiento, pruebas diagnósticas, resultados y un último apartado aplicable exclusivamente a usuarios que acuden al servicio de emergencias sin haber

sido derivados por un médico [1]. En la Tabla 2.2. se detallan los criterios y características que deben considerarse en su aplicación.

Tabla 2.2. Protocolo de Adecuación de Emergencias y/o Urgencias Hospitalarias

Protocolo de adecuación de Emergencias y/o Urgencias hospitalarias
1) CRITERIOS DE GRAVEDAD
1.1 Pérdida de conciencia, desorientación, coma, insensibilidad (brusca o muy reciente)
1.2 Pérdida brusca de visión o audición
1.3 Alteraciones pulso (<50/>140 pulsaciones/minuto) y arritmias.
1.4 Alteración presión arterial (sistólica: <90/>200 mmHg; diastólica: <60/>120 mmHg)
1.5 Alteraciones de electrolitos o gases sanguíneos (No considerar en usuarios con alteraciones crónicas de estos parámetros: insuficiencia renal crónica, insuficiencia respiratoria crónica, etc.)
1.6 Fiebre persistente (5 días) no controlada con tratamiento en atención primaria.
1.7 Hemorragia activa (hematemesis, epíxtasis, melenas, etc.). Excluye heridas superficiales que sólo requieren sutura.
1.8 Perdida brusca de la capacidad funcional de cualquier parte del cuerpo
2) CRITERIOS DE TRATAMIENTO.
2.1 Administración de medicación o fluidos por vía intravenosa (excepto mantenimiento de vía)
2.2 Administración de oxigeno
2.3 Colocación de yesos (excluye vendajes)
2.4 Intervención/procedimiento realizado en quirófano.
3) CRITERIOS DE INTENSIDAD DIAGNOSTICA
3.1 Monitorización de signos vitales o toma de constantes cada 2 horas.
3.2 Radiología de cualquier tipo.
3.3 Pruebas de laboratorio (excepto glucemia en diabéticos que acuden por motivos no relacionados con la diabetes y pruebas de glucemia en tira seca).
3.4 Electrocardiograma (excepto cardiopatía crónica que acude por problemas no relacionados cardiopatía)
4) OTROS CRITERIOS
4.1 El usuario está más de 12 horas en observación en Emergencias
4.2 El usuario es ingresado en el hospital o trasladado a otro hospital
4.3 El usuario fallece en Emergencias
4.4 Otros en usuario derivado por un médico (especificar).
5) CRITERIOS APLICABLES SÓLO A USUARIOS QUE ACUDEN ESPONTANEAMENTE
5.1 Proviene de un accidente (tráfico, laboral, en lugar público,...) y hay que evaluar al usuario.
5.2 Sintamos que sugieren urgencia vital: dolor torácico, disnea instauración rápida, tiraje, dolor abdominal agudo
5.3 Cuadro conocido por el usuario y que habitualmente requiere ingreso.
5.4 Al usuario le ha dicho un médico que acuda a Urgencias si se presenta el síntoma
5.5 Requiere atención medica primaria con rapidez y el hospital es el centro más cercano
5.6 Otros en usuarios espontáneos (especificar).

NOTA: Los detalles para tomar en cuenta para la atención en salas de emergencia según el Ministerio de Salud y Deporte. [1]

Estas son algunas de las tecnologías presentes en las salas de emergencia en Bolivia, lo que claramente evidencia una carencia de innovación y avance.

2.5. Retos actuales

Los retos actuales son múltiples y abarcan desde lo humano hasta lo organizacional. Uno de los problemas más señalados es la escasez de personal capacitado en medicina de emergencias, ya que la mayoría de los profesionales que atienden en estas áreas son médicos generales sin especialización específica [15].

Un aspecto crítico es el tiempo promedio de espera para ser atendido, que puede superar las dos horas en muchos hospitales, incluso en casos moderadamente urgentes.

Además, muchas salas de emergencia carecen de tecnología básica como monitores multiparámetro, desfibriladores funcionales o sistemas informatizados de historia clínica, lo cual compromete tanto la seguridad como la eficiencia de la atención [16].

Todos estos obstáculos e inconvenientes son analizados y tomados en cuenta para la elaboración del asistente para diagnóstico médico que, si bien no representa una solución directa a las necesidades requeridas por la población, ayuda a aliviar los malestares de los pacientes y mejora la atención en las salas de emergencia.

CAPÍTULO 3

MODELOS DE INTELIGENCIA ARTIFICIAL

3. MODELOS DE INTELIGENCIA ARTIFICIAL

3.1. Caracterización de la inteligencia artificial

3.1.1. Definición

Hoy en día, se puede dar por hecho que una de las tecnologías más influyentes alrededor del mundo es la inteligencia artificial (IA) ya que cuenta con aplicaciones para una diversa cantidad de campos como la industria, educación, comercio e incluso el área de la salud. La Inteligencia Artificial o IA es una rama de la informática que se centra en el desarrollo de sistemas informáticos capaces de realizar tareas que habitualmente requieren de inteligencia humana [17].

La IA ayuda a simplificar y automatizar procesos (tareas repetitivas, largas o complicadas en las que la robótica puede reemplazar las acciones humanas) y libera tiempo a los empleados para que puedan dedicarlo a otras tareas que aporten valor añadido [17].

3.1.2. Funcionamiento

Para que la Inteligencia Artificial funcione necesita datos, por este motivo, el desarrollo de esta tecnología ha ido de la mano de otras como Big Data, Internet de las Cosas o la red móvil 5G. Es esa información la que permite crear patrones a las máquinas y sistemas (softwares) con los que aprender [18]. Si bien los detalles varían según las diferentes técnicas de IA, el principio central gira en torno a los datos. Los sistemas de IA aprenden y mejoran a través de la exposición a grandes cantidades de datos.

Este proceso de aprendizaje suele implicar algoritmos, que son conjuntos de reglas o instrucciones que guían el análisis y la toma de decisiones de la IA. En el aprendizaje automático, un subconjunto popular de la IA, los

algoritmos se entrenan con datos etiquetados o no etiquetados para hacer predicciones o categorizar información [19].

3.1.3. Tipos de inteligencia artificial

La inteligencia artificial (IA) se clasifica según el nivel de inteligencia que presenta, con esto en mente, se encuentran los siguientes tipos:

- **IA estrecha:** La inteligencia artificial estrecha (ANI) también se conoce como IA débil. Este tipo de IA tiene un rango limitado de capacidades y se enfoca en una sola tarea. La mayoría de las aplicaciones de la IA que se usan en la vida cotidiana entran en esta clasificación [20].
- **IA general:** La inteligencia artificial general (AGI) se trata de una inteligencia capaz de imitar las funciones cerebrales más complejas. Aún está en proceso de desarrollo, pero se espera que en el futuro puedan alcanzar capacidades humanas [20].

3.2. Modelos aplicables de inteligencia artificial

Los modelos ayudan a entender las distintas formas en las que puede funcionar la inteligencia artificial. A continuación, se detallan los modelos de inteligencia artificial que son aplicables para la implementación.

3.2.1. Sistemas basados en reglas

Los sistemas basados en reglas son una de las formas más simples de IA. Funcionan mediante conjuntos de instrucciones tipo "si ocurre esto, entonces haz aquello", y son útiles en entornos donde las decisiones pueden predefinirse con precisión. Sin embargo, presentan limitaciones importantes, como su incapacidad para adaptarse a contextos nuevos o no previstos [21].

3.2.2. Redes neuronales artificiales

Inspiradas en el funcionamiento del cerebro humano, las redes neuronales están compuestas por capas de nodos (neuronas artificiales) interconectadas. Estas redes aprenden ajustando los pesos de las conexiones internas a medida que procesan grandes volúmenes de datos. Son especialmente eficaces en tareas como el reconocimiento de imágenes, voz y texto [22].

3.2.3. Algoritmos de aprendizaje automático (ML)

Los algoritmos de aprendizaje automático, también conocidos como machine learning, constituyen una rama fundamental dentro de la inteligencia artificial. Estos algoritmos permiten que los sistemas informáticos aprendan de los datos, identifiquen patrones y tomen decisiones con mínima o nula intervención humana. En lugar de seguir una secuencia de instrucciones explícitas, el sistema ajusta su comportamiento a medida que recibe nueva información, mejorando su precisión con el tiempo [21]. Se suelen dividir en 3 grandes categorías.

3.2.3.1. Aprendizaje Supervisado

El aprendizaje supervisado consiste en entrenar algoritmos a partir de un conjunto de datos etiquetados, es decir, aquellos que incluyen tanto la entrada como la salida deseada. El modelo aprende la relación entre ambas variables para luego predecir resultados sobre datos nuevos [23].

3.2.3.2. Aprendizaje no supervisado

A diferencia de las técnicas de aprendizaje supervisado, el aprendizaje no supervisado no supone la existencia externa de respuestas “correctas” o “incorrectas” y, por lo tanto, no requiere etiquetado. Estos algoritmos

detectan patrones inherentes en conjuntos de datos para agrupar puntos de datos e informar predicciones [21].

3.2.3.3. Aprendizaje por esfuerzo

En el aprendizaje por refuerzo, un modelo aprende de manera holística mediante ensayo y error, y recompensa sistemáticamente los outputs correctos (o penaliza los incorrectos). Los modelos de refuerzo se utilizan en las sugerencias de las redes sociales, la negociación algorítmica de acciones e incluso en los vehículos autónomos [21].

3.2.4. Modelo de aprendizaje profundo

El aprendizaje profundo es un subconjunto especializado del aprendizaje automático que imita el funcionamiento del cerebro humano utilizando redes neuronales artificiales. El aprendizaje profundo es particularmente adecuado para aplicaciones que requieren reconocimiento de patrones complejos, como el reconocimiento de imágenes y audio, así como el procesamiento del lenguaje natural [24].

3.2.5. Modelo de NLP

El procesamiento de lenguaje natural (NLP) combina lingüística e informática para centrarse en la interacción entre las computadoras y el lenguaje humano. Su objetivo es dotar a las máquinas de la capacidad para entender, interpretar y manipular el lenguaje humano, permitiendo que las computadoras comprendan y generen texto o voz de forma similar a como lo harían las personas [25].

3.3. Inteligencia artificial en la medicina

Los últimos avances tecnológicos han facilitado la rápida incorporación de la inteligencia artificial (IA) a la atención médica contemporánea, impulsando

algunos de los progresos más significativos en el ámbito de la salud en los últimos años [26].

3.3.1. Impacto en el área de la medicina

La inteligencia artificial ha tenido un impacto sin precedentes en el sector de la salud. Gracias a sus numerosas funcionalidades, la inteligencia artificial en medicina se ha extendido desde el área asistencial, hasta el campo de la educación sanitaria [27].

Actualmente, las funciones más comunes de la IA en entornos médicos son el apoyo a la toma de decisiones clínicas y el análisis de imágenes. Las herramientas de apoyo a la toma de decisiones clínicas ayudan a los proveedores a tomar decisiones sobre tratamientos, medicamentos, salud mental y otras necesidades de los pacientes, proporcionándoles un acceso rápido a información o investigaciones relevantes para sus pacientes. En las imágenes médicas, las herramientas de IA se utilizan para analizar radiografías, resonancias magnéticas y otras imágenes en busca de lesiones u otros hallazgos que un radiólogo humano podría pasar por alto [21].

Por otro lado, tenemos otras aplicaciones como:

- Prevención de enfermedades
- Ayuda en la toma de decisiones médicas
- Diagnóstico y tratamiento de enfermedades
- Formación para profesionales sanitarios
- Investigación

3.3.2. Modelos aplicados al diagnóstico médico

En la práctica clínica, los modelos de inteligencia artificial que más se emplean en el diagnóstico médico son el aprendizaje supervisado, las redes

neuronales artificiales y el aprendizaje profundo (deep learning), así como el procesamiento de lenguaje natural (NLP). Estos modelos han demostrado gran eficacia en diferentes contextos médicos, especialmente en el análisis de imágenes, la predicción de enfermedades y la gestión de información clínica.

El aprendizaje supervisado y las redes neuronales artificiales son ampliamente utilizados en la interpretación de imágenes médicas, ya que permiten entrenar modelos con datos previamente etiquetados. Un ejemplo es el uso de redes neuronales convolucionales (CNNs) para la detección de neumonía y COVID-19 en radiografías y tomografías computarizadas, alcanzando altos niveles de precisión diagnóstica [28].

También, el procesamiento de lenguaje natural (NLP) se ha convertido en una herramienta valiosa para el análisis de información no estructurada en los registros médicos electrónicos. Este enfoque permite extraer automáticamente entidades clínicas como síntomas, diagnósticos o procedimientos a partir de notas médicas, lo que contribuye a mejorar la exactitud y eficiencia en la práctica clínica [29].

3.3.3. Consideraciones éticas y legales de la inteligencia artificial en la medicina

La inteligencia artificial (IA) ha revolucionado la medicina moderna al ofrecer herramientas que apoyan el diagnóstico, tratamiento y seguimiento de pacientes. Sin embargo, su aplicación plantea importantes retos éticos y legales que deben ser atendidos para garantizar su uso responsable y seguro.

Desde el punto de vista ético, uno de los principales desafíos es asegurar que el uso de IA respete principios fundamentales como la autonomía del paciente, la justicia, la beneficencia y la no maleficencia. Por ejemplo, los

pacientes deben ser informados de manera clara sobre el uso de IA en sus tratamientos, y se debe garantizar que esta tecnología no reproduzca sesgos que puedan afectar negativamente a ciertos grupos poblacionales [30].

Otro aspecto clave es la responsabilidad legal. A pesar de que las decisiones clínicas asistidas por IA pueden ser más eficientes, todavía no existe un marco normativo uniforme que determine quién es responsable en caso de errores. En general, la responsabilidad recae en los profesionales de la salud, aunque el desarrollo autónomo de algunos sistemas genera dudas sobre cómo compartir esa carga legal [31].

Ante estos desafíos, distintos organismos han propuesto marcos éticos para guiar el desarrollo y uso de la IA en salud. En América Latina, por ejemplo, se impulsa un enfoque que promueva la equidad, la inclusión y el respeto a los derechos humanos en el contexto digital [32]. Si bien la IA representa una herramienta poderosa para mejorar la medicina, su implementación debe realizarse con criterios éticos y legales claros, priorizando la seguridad de los pacientes y el respeto por sus derechos.

CAPÍTULO 4

DISEÑO DEL ASISTENTE PARA DIAGNÓSTICO MÉDICO

4. DISEÑO DEL ASISTENTE PARA DIAGNÓSTICO MÉDICO

4.1. Descripción del proceso

El prototipo desarrollado consiste en un sistema que permite al paciente ingresar información en una pantalla donde detalla sus síntomas y algunos antecedentes médicos relevantes, tales como intervenciones quirúrgicas previas, enfermedades preexistentes o alergias a determinados medicamentos. También, mediante sensores se toman los signos vitales del paciente para hacer un diagnóstico aún más exacto y así estar más al tanto del estado del paciente. Cuando se ingresa esta información, el asistente procesa los datos mediante inteligencia artificial y genera una respuesta que incluye una estimación del nivel de gravedad del caso y recomendaciones preliminares.

Es importante señalar que las recomendaciones proporcionadas por el asistente tienen como finalidad aliviar temporalmente los síntomas del paciente mientras se espera atención médica. El asistente no sustituye la evaluación de un profesional de la salud ni pretende reemplazar su criterio clínico. Adicionalmente, la clasificación de la gravedad del caso tiene como objetivo asistir al personal médico en la priorización de pacientes y facilitar la atención de aquellos que se encuentran en condiciones más críticas. Esta funcionalidad resulta especialmente útil en contextos hospitalarios públicos, donde los tiempos de espera suelen ser prolongados.

4.2. Instrumentación requerida

Una vez descrito el funcionamiento del asistente, se procede con la evaluación de los instrumentos y materiales requeridos para su diseño, elaboración e implementación.

4.2.1. Raspberry Pi 4

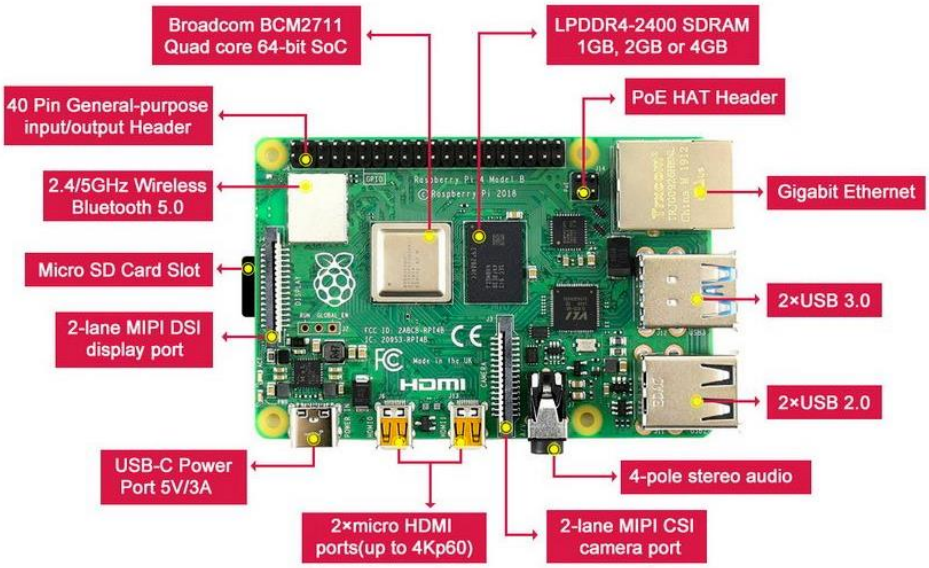
El computador es la parte más esencial del asistente ya que es el cerebro de todo, el que se encarga de procesar toda la información. Para esta ocasión se escoge la Raspberry Pi 4 Modelo B de 4GB de RAM. En la Ilustración 4.1. se puede ver el computador que se utiliza.

Ilustración 4.1. Raspberry Pi Modelo B

NOTA: Presenta mayor velocidad del procesador y mejor rendimiento frente a otras opciones en el mercado [33].

Se utiliza este dispositivo debido a la facilidad de conectividad que tiene. Al ser compatible con una gran cantidad de otros componentes y dispositivos, se amplían las opciones aprovechables para desarrollar el prototipo. Los 4GB de RAM que incluye son adecuados ya que el asistente no requiere de procesos grandes o pesados. La Raspberry Pi 4 cuenta con un procesador de 1.5GHz, soporta doble monitor (4Kp60) con salida en dos conectores micro HDMI, USB 3.0, WiFi doble banda 2.4GHz y 5GHz, Bluetooth 5.0 y puerto Ethernet [34]. En la Ilustración 4.2. se muestran todos los conectores que tiene el dispositivo.

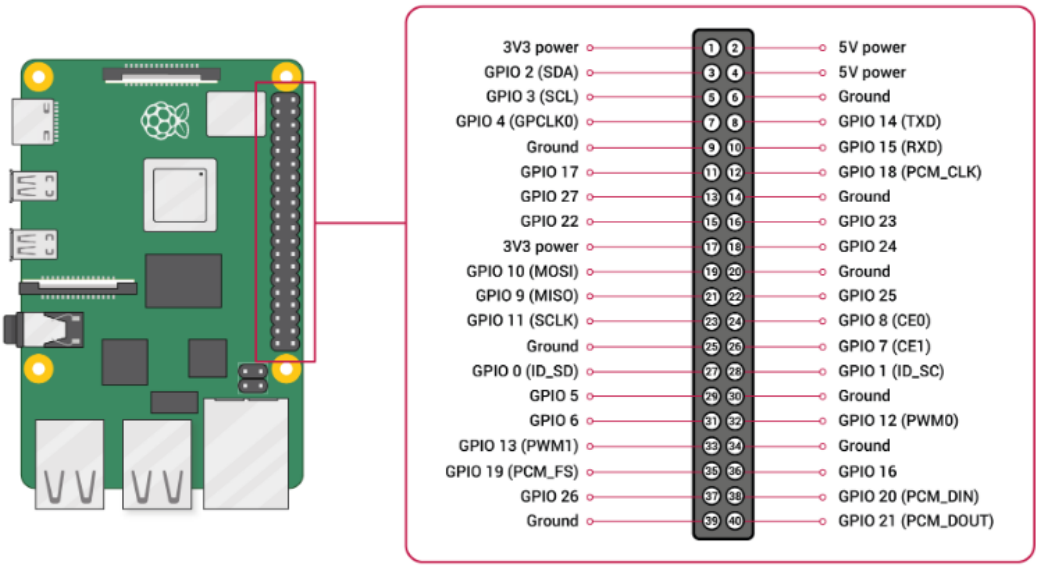
Ilustración 4.2. Diagrama de conexiones de la Raspberry Pi 4



[35]

En la Ilustración 4.3. se pueden ver los pines GPIO; son pines de propósito general que permiten comunicar y controlar dispositivos externos para luego establecer ya sea entrada o salida [36].

Ilustración 4.3. Pines GPIO de la Raspberry Pi 4 Modelo B



[35]

4.2.1.1. Sistema operativo

Cuando se hace uso de cualquier tipo de Raspberry Pi, es necesario instalar un sistema operativo para que funcione. Para este caso se instala Raspberry Pi OS Lite de 32-bit sin entorno gráfico. Se utiliza este sistema operativo ya que es muy liviano a comparación de los demás, lo que ayuda a que no se sobrecaliente la Raspberry Pi, además, al no tener entorno gráfico, se puede administrar mejor el espacio y el rendimiento del sistema, instalando solo lo necesario y no sobrecargar tanto el almacenamiento.

4.2.1.2. Programación

La programación se utiliza en todas las etapas del desarrollo del asistente. En esta ocasión se realiza en lenguaje Python. Luego de descargar el sistema operativo, se utiliza Python para instalar algunos recursos y actualizaciones necesarias desde la terminal del sistema operativo. Se instala un entorno grafico básico y con Python se empieza a programar toda la interfaz gráfica para que funcione el asistente de manera correcta. En la programación se asignan los parámetros necesarios, entradas y salidas del sistema y los requerimientos para un buen funcionamiento general.

4.2.2. Pantalla táctil HMTECH 10.1

La pantalla es una parte importante ya que funciona como interfaz de entrada para la información del paciente. En esta ocasión se opta por una pantalla táctil HMTECH de 10.1 pulgadas. Esta pantalla está desarrollada y enfocada para trabajar con Raspberry Pi lo que facilita su implementación. Se trata de una pantalla capacitiva de alta resolución con un táctil preciso y fácil de usar, ya que solo se conecta el cable HDMI al puerto correspondiente y otro cable USB para habilitar la opción táctil de la pantalla. En la Ilustración 4.4. se muestra la pantalla que se utiliza.

Ilustración 4.4. Pantalla táctil HMTECH 10.1

NOTA: La pantalla tiene tecnología IPS lo que brinda una mejor resolución a diferentes ángulos [37].

4.2.3. Sensor de temperatura corporal MLX90614

Uno de los parámetros más relevantes en el área de la salud es la temperatura corporal, esta ayuda a determinar ciertos indicios de infección o inflamación en el paciente. El uso de sensor de temperatura corporal en el asistente sirve para realizar una evaluación más rápida y precisa, debido a esto, se elige el sensor MLX90614. En la Ilustración 4.5. se muestra el sensor que se utiliza.

Ilustración 4.5. Sensor para medir temperatura corporal MLX90614

NOTA: El sensor que se elige es de fácil implementación en proyectos de electrónica [38].

Una de las principales ventajas del MLX90614 es su capacidad para realizar mediciones sin necesidad de contacto físico, lo cual lo hace especialmente adecuado para el monitoreo de temperatura corporal en entornos clínicos donde se busca evitar el riesgo de contagio o contaminación cruzada.

4.2.4. Sensor para latidos del corazón COMIMARK

Al igual que la temperatura corporal, el monitoreo del ritmo cardiaco es un componente esencial en el diagnóstico médico de un paciente. Para este caso se opta por el Pulse Heart Rate Sensor COMIMARK. Se usa este sensor debido a su accesibilidad y a su tamaño, ya que, al no tener dimensiones tan grandes, se puede adaptar a varios espacios. En la Ilustración 4.6. se muestra el sensor que usa el asistente.

Ilustración 4.6. Pulse Heart Rate Sensor COMIMARK

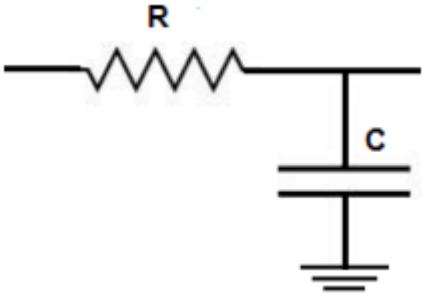
NOTA: Usa tecnología de fotopletismografía (PPG), mide los cambios en el volumen sanguíneo venoso [39].

Un detalle que presenta este sensor es que el valor de salida es una señal analógica, señal que no puede ser procesada por los pines GPIO de la Raspberry Pi, por lo que para que sea procesada hace falta filtrar la señal y convertir a una señal digital.

4.2.4.1. Filtro paso bajo

Un filtro paso bajo permite el paso de señales por debajo de una frecuencia de corte (banda de paso) y atenúa señales por encima de la frecuencia de corte (banda de atenuación) [40]. Se decide hacer uso de este elemento con el objetivo de eliminar el ruido de alta frecuencia que puede aparecer al medir los latidos de corazón del paciente. Esto se genera debido a algún movimiento indeseado del paciente o a algún factor externo que puede hacer que la frecuencia se altere y no dar una medición exacta. Para evitar eso, se realiza una configuración como se muestra en la Ilustración 4.7.

Ilustración 4.7. Filtro paso bajo



NOTA: Circuito para la configuración de un filtro paso bajo [41].

4.2.4.2. Convertidor ADC MCP3008

Para que el asistente procese las señales del sensor, hace falta un convertidor ADC. Este dispositivo convierte una señal analógica en una señal digital. Esto es necesario en el sistema ya que se requiere que las señales físicas se conviertan en datos digitales para pueda ser procesado por los pines GPIO de la Raspberry Pi. Otros sensores normalmente tienen integrado todo este sistema de manera interna, por lo que no siempre hace falta hacer uso de un convertidor aparte, sin embargo, el sensor de pulso COMIMARK no viene integrado con un convertidor por lo que en este caso si es requerido. En este

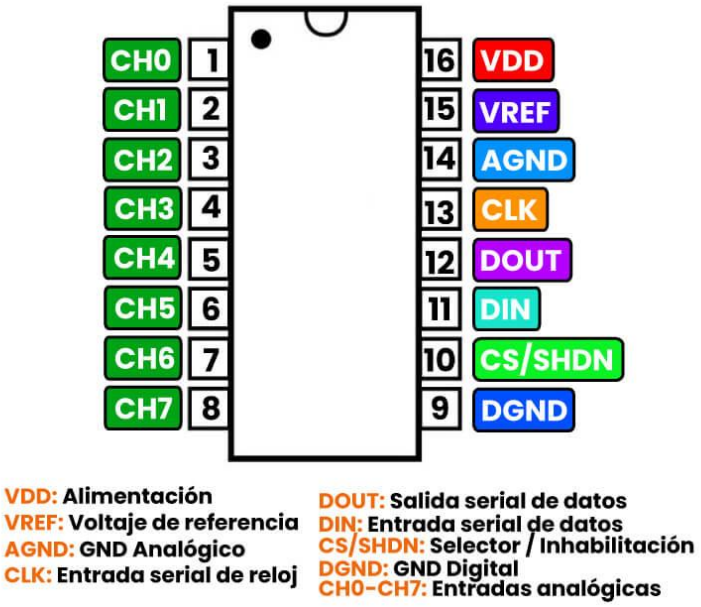
caso se opta por el convertidor ADC MCP3008 ya que es el que se adecua más al prototipo. En la Ilustración 4.8. se detalla el dispositivo.

Ilustración 4.8. Convertidor ADC MCP3008

NOTA: Convertidor de señales analógicas a digitales [42].

El convertidor se conecta a la Raspberry Pi a través de los pines GPIO. En la Ilustración 4.9. se detallan los conectores del convertidor.

Ilustración 4.9. Pines del convertidor ADC MCP3008

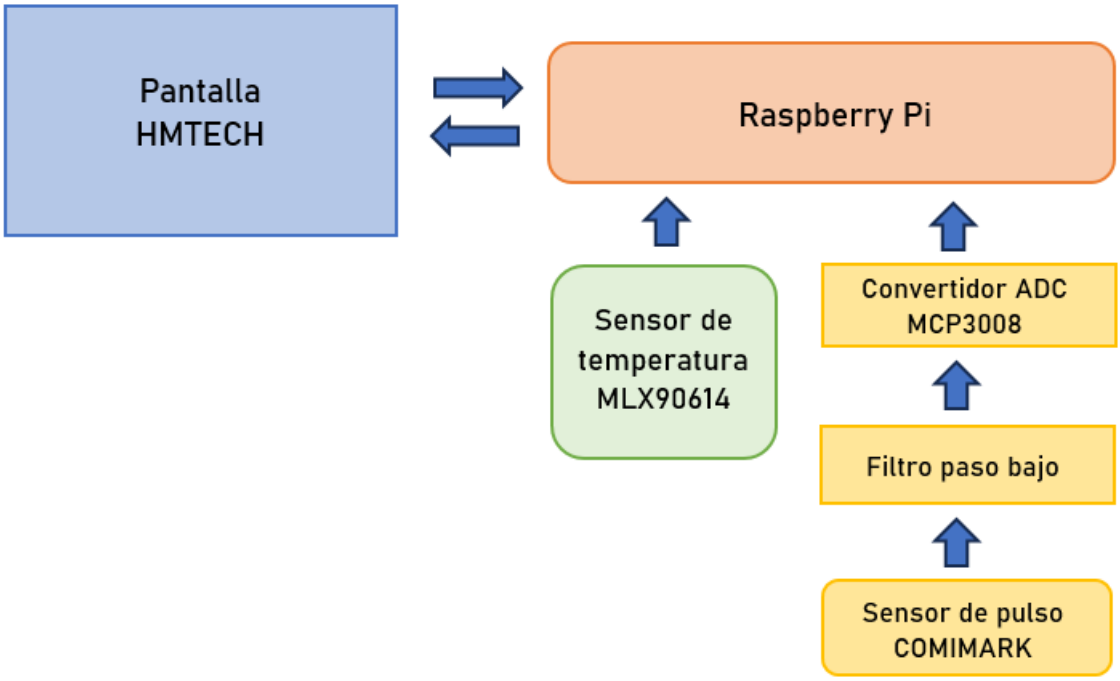


NOTA: En este caso, la entrada viene del filtro paso bajo que se menciona anteriormente y la salida del convertidor se conecta a los pines GPIO de la Raspberry Pi [42].

4.3. Análisis de conexiones

En esta sección, se analizan las conexiones que se realizan para el funcionamiento del asistente. Primero, se definen la entradas y salidas del sistema. Como dispositivos de entrada se tiene a los sensores tanto de temperatura como de latidos del corazón y también a la pantalla táctil, ya que en esta se ingresa información relevante para el análisis del paciente. Por otro lado, como dispositivos de salida solo se tiene a la pantalla, ya que en ella se muestran los resultados de la evaluación hecha por el asistente. En la Ilustración 4.10. se detalla las conexiones de todos los dispositivos.

Ilustración 4.10. Diagrama de conexiones



NOTA: En el diagrama se muestran todos los dispositivos y sus conexiones correspondientes.

En la Raspberry Pi se conectan todos los dispositivos. Para la conexión de la pantalla, se conecta el cable HDMI desde el puerto de la pantalla hasta el puerto HDMI del computador. Para el sensor de temperatura MLX90614 se

conectan sus cables a los puertos GPIO correspondientes. Para el sensor de pulso COMIMARK se conectan los cables del sensor a la entrada del filtro paso bajo, luego, la salida del filtro se conecta al puerto de entrada del convertidor ADC MCP3008, y, por último, la salida del convertidor se conecta a los puertos GPIO de la Raspberry Pi.

4.4. Diseño del asistente

El diseño del asistente se divide en la parte de hardware, donde se realiza el armado físico y las conexiones de dispositivos, y en la parte de software, donde se incluye la programación y la integración de todas las partes.

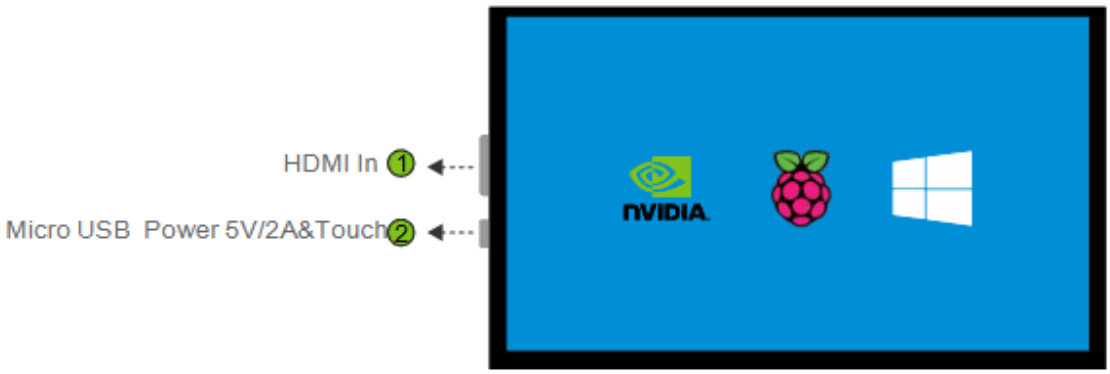
4.4.1. Hardware

En la parte de hardware se detalla el diseño del sistema en general, el diseño de los circuitos necesarios y el diseño de la estructura física del asistente.

4.4.1.1. Conexiones del sistema

La pantalla táctil HMTECH cuenta con un puerto HDMI y un puerto mini USB. En la Ilustración 4.11. se detallan los conectores del dispositivo.

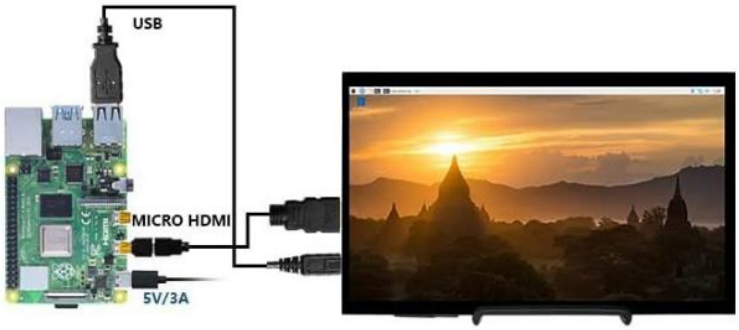
Ilustración 4.11. Conectores físicos de la pantalla HMETECH



NOTA: Solo presenta dos conectores para transmitir imagen, activar el táctil y para la alimentación de la pantalla [43].

La Raspberry Pi cuenta con conectores HDMI mini para conectar la pantalla, además de puertos USB para conectar el cable mini USB desde la pantalla. En este caso, se usa un adaptador para conectar al puerto HDMI mini. En la Ilustración 4.12. se muestra la conexión de la pantalla con la Raspberry Pi.

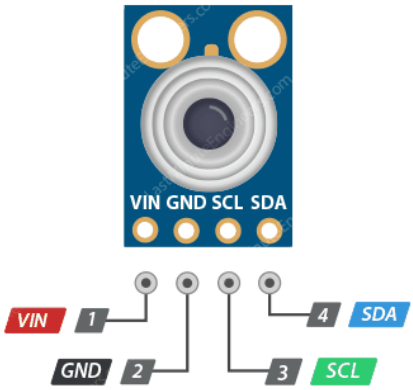
Ilustración 4.12. Conexión entre la Raspberry Pi y la pantalla HMTECH



NOTA: Se detalla la alimentación de la Raspberry Pi que se realiza con su respectivo cable [43].

Por otro lado, se tienen los sensores. Primero, el sensor para medición de temperatura MLX90614 cuenta con 4 pines para las conexiones con la Raspberry Pi. En la Ilustración 4.13. se detallan los pines del sensor.

Ilustración 4.13. Pines para conectar el sensor MLX90614

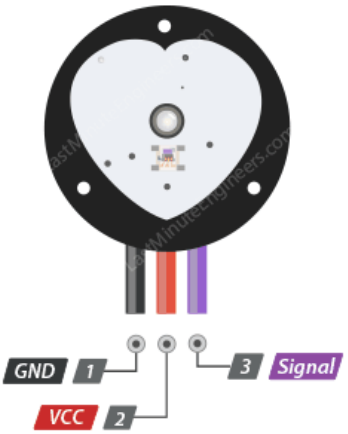


NOTA: Estos pines no requieren alguna conexión extra, por lo que se conectan directamente a la Raspberry Pi [44].

Los conectores VIN y GND son para la alimentación del sensor y se conectan directamente a los pines de GND y 5V de la Raspberry Pi. Para los otros pines, el SCL es el reloj del sensor, marca el ritmo de la comunicación, y el SDA sirve para transportar la información registrada. Se conectan a los pines GPIO 3 para SCL y GPIO 2 para SDA.

El sensor para medición de pulso cuenta con 3 pines. En la Ilustración 4.14. se detalla cada uno de ellos.

Ilustración 4.14. Pines para medición de pulso COMIMARK



NOTA: Los pines de alimentación se conectan directo a la Raspberry Pi mientras que la señal hay que convertirla a digital [44].

Los pines de alimentación se conectan directamente a los pines de 5V y GND en la Raspberry Pi. La señal de salida del sensor es una señal analógica, para que pueda ser procesada debe convertirse a una señal digital por lo que se aplica un filtro y un convertidor ADC.

El filtro elimina ruido de alta frecuencia o posibles interferencias que se generan por algún movimiento del paciente al hacer la medición. Antes de eso, se diseña el filtro y se realizan los cálculos necesarios para saber los valores adecuados de los componentes que requiere el asistente.

El filtro que se utiliza es un filtro activo paso bajo de segundo orden tipo Sallen-Key. Se usa este tipo de filtro ya que preserva la forma original de la señal cardiaca mientras elimina eficazmente las interferencias. El filtro paso bajo se encarga de limitar las frecuencias permitidas y la arquitectura tipo Sallen-Key representa el equilibrio ideal entre rendimiento y practicidad: al ser de segundo orden proporciona una atenuación significativa del ruido de alta frecuencia mientras mantiene una implementación simple con componentes fácilmente disponibles. Al ser un filtro activo, el uso de un amplificador operacional integrado mejora la calidad de la señal y asegura la integración entre etapas, ya que proporciona una salida compatible con el convertidor ADC MCP3008.

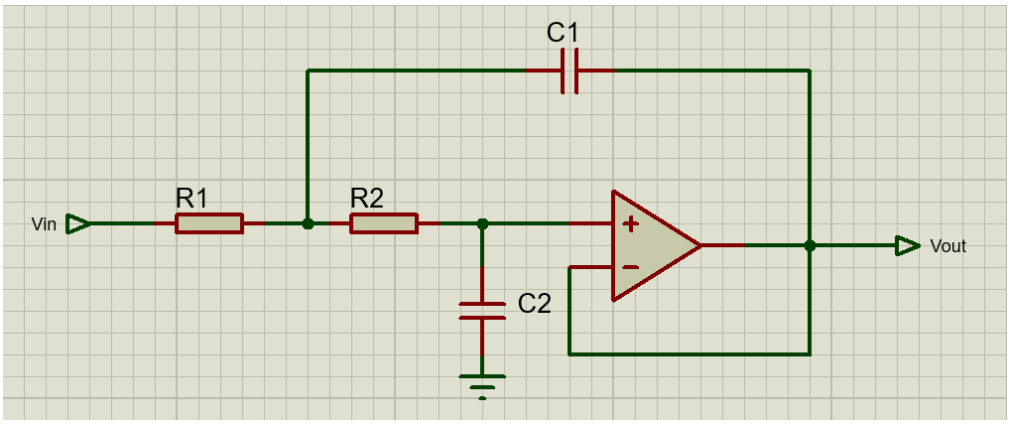
Para su diseño, primero se establece la frecuencia de corte (fc) que corresponde al rango en el cual se va a trabajar. Para una persona en reposo, los latidos del corazón son de 60 bpm a 100 bpm [45]. También, en algunas ocasiones, los pacientes pueden llegar con bradicardia (40 bpm) o con taquicardia (180 bpm). Para abarcar todos estos valores, se decide trabajar con un rango que llega hasta los 240 bpm, de esta manera, se abarcan todos los casos posibles.

Ahora, se realiza la conversión de bpm a Hertz (Hz), se toma en cuenta que 1 Hz equivale a 60 bpm.

$$fc = 240\ bpm = \frac{1\ Hz}{60\ bpm} = 4\ Hz$$

La frecuencia de corte es de 4 Hz, entonces todas las señales por encima se van a atenuar. La arquitectura Sallen-Key establece una configuración que se debe seguir y calcular. En la Ilustración 4.15. se muestra el circuito que corresponde a la arquitectura del filtro que se utiliza.

Ilustración 4.15. Arquitectura tipo Sallen-Key de segundo orden



NOTA: La ganancia es unitaria. Fuente: Elaboración propia

Ahora se analizan las ecuaciones que pertenecen a este tipo de arquitectura.

$$Q = \frac{1}{3-K} \qquad (1)$$

$$fc = \frac{1}{2\pi\sqrt{R_1 R_2 C_1 C_2}} \qquad (2)$$

En la ecuación (1) se calcula el factor de calidad Q donde K=1 ya que la ganancia con la que se trabaja es unitaria.

$$Q = \frac{1}{3-K} = \frac{1}{3-1} = \frac{1}{2} = 0.5$$

El Q = 0.5 indica que se aplica la configuración Butterworth. Esto garantiza que el filtro mantenga la señal del pulso intacta, elimina el ruido y no distorsiona la información. Luego se trabaja con la ecuación (2), el Q = 0.5 permite hacer la siguiente relación para simplificar.

$$R_1 = R_2 = R \qquad C_1 = C_2 = C$$

Se reemplaza la ecuación (2) con la relación anterior.

$$fc = \frac{1}{2\pi\sqrt{R^2C^2}} = \frac{1}{2\pi RC} \qquad (3)$$

Para obtener los valores de los componentes se despeja la ecuación para aproximar a los valores disponibles en el mercado.

$$fc * 2\pi * RC = 1$$

$$C = \frac{1}{2\pi fc} * R \qquad (4)$$

Se asigna un valor a R para poder calcular el valor de C. Para esta ocasión, se trabaja con resistencias de $100k\Omega$. Luego, se reemplaza en la ecuación (4) para obtener un resultado para C.

$$C = \frac{1}{2\pi(4\ Hz)}(100k\Omega) = 3.98\ x\ 10^{-7} = 0.398\ \mu F$$

Se compara con los valores comercialmente posibles de capacitores y se obtienen las siguientes opciones: $0.33\ \mu F$ y $0.47\ \mu F$. Ahora se reemplazan esos valores en la ecuación (3) para ver cuál de ambos valores se aproximan más a la frecuencia de corte (4 Hz).

Para $0.33\ \mu F$: $\quad fc = \frac{1}{2\pi RC} = \frac{1}{2\pi(100k\Omega)(0.33\mu F)} = 4.82\ Hz$

Para $0.47\ \mu F$: $\quad fc = \frac{1}{2\pi RC} = \frac{1}{2\pi(100k\Omega)(0.47\mu F)} = 3.39\ Hz$

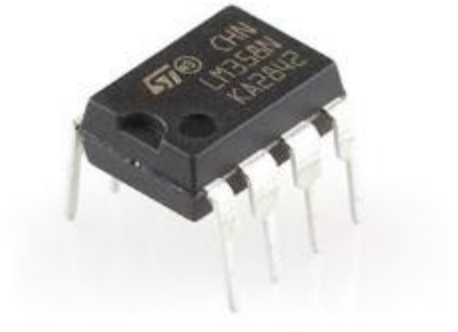
Entonces, se opta por usar un capacitor de $0.33\ \mu F$ ya que es el componente con mayor disponibilidad en el mercado. Ahora, se tienen los componentes a utilizar con los valores correspondientes.

$$R_1 = R_2 = R = 100\ k\Omega$$

$$C_1 = C_2 = C = 0.33\ \mu F$$

Para completar el diseño del filtro, se escoge el amplificador operacional LM358. En la Ilustración 4.16. se detalla el amplificador operacional.

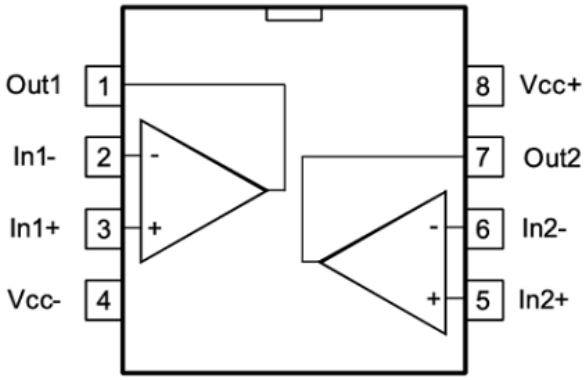
Ilustración 4.16. Componente LM358



NOTA: Este componente cuenta con 2 amplificadores internamente [46].

Se escoge este componente en específico ya que el componente puede trabajar con 5 V y la Raspberry Pi tiene un pin que suministra esa cantidad, por lo que no hace falta añadir otro tipo de alimentación. En la Ilustración 4.17. se detallan los pines que tiene el componente.

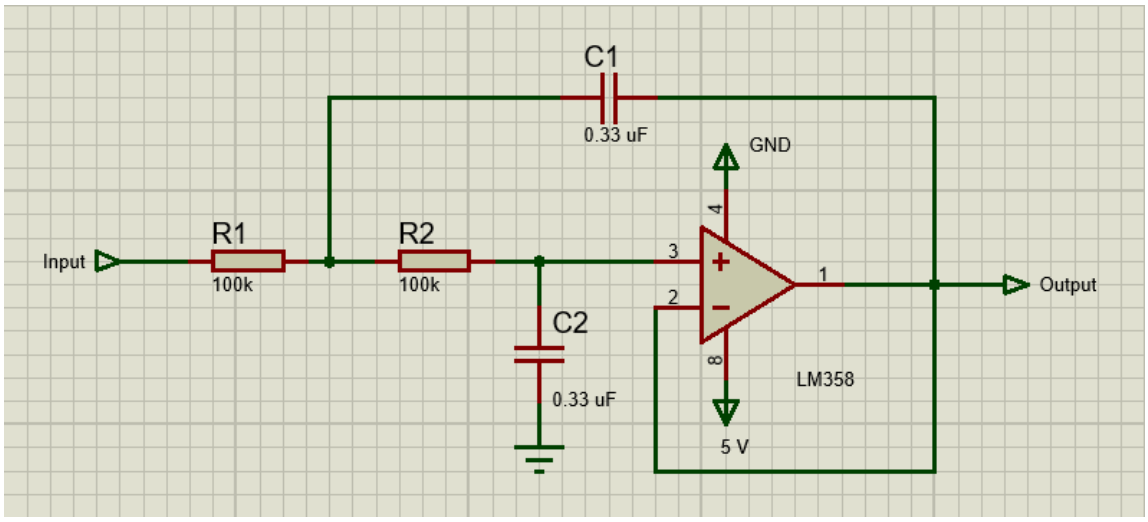
Ilustración 4.17. Pines del LM358



NOTA: En este caso, solo se hace uso de un amplificador operacional [46].

Una vez que se identifican los pines a utilizar, primero se realiza la conexión para alimentar el componente mediante los pines de Vcc+ y Vcc-. Luego se conectan los componentes previamente calculados. En la Ilustración 4.18. se detalla la configuración final de todo el filtro activo paso bajo de segundo orden tipo Sallen-Key.

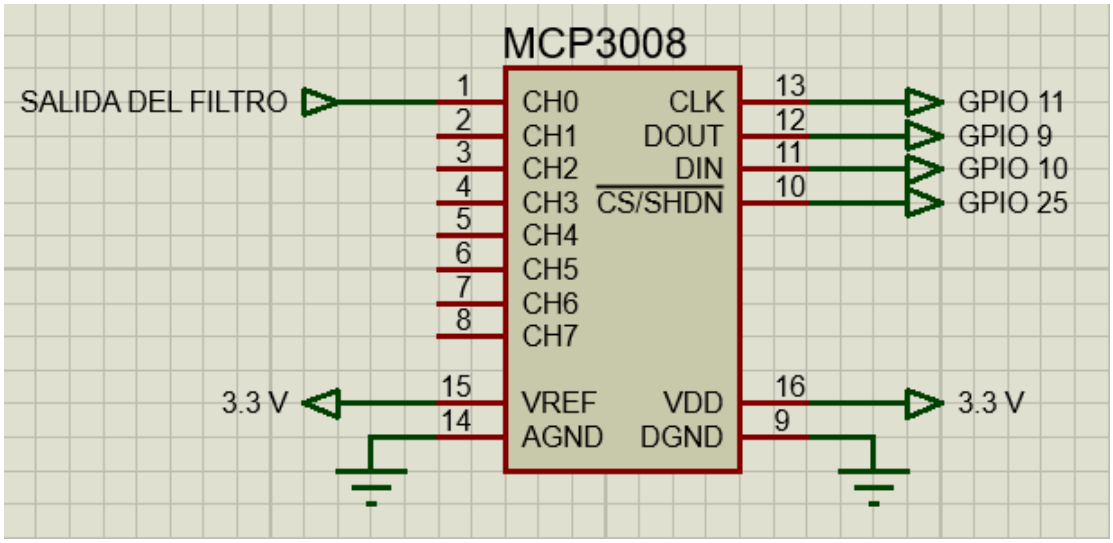
Ilustración 4.18. Diseño del filtro completo



NOTA: En el filtro se detallan los valores de las resistencias, los capacitores y el amplificador LM358.

Una vez se diseña el filtro, la salida de este se conecta a la entrada del convertidor ADC MCP3008. Existe una discrepancia en los rangos de voltaje entre el filtro y el MCP3008. La salida del filtro activo presenta un margen de salida limitado, operando aproximadamente entre 0.1 V y 3.5 V. Por otro lado, el MCP3008, se alimenta hasta con 5 V, tiene un rango de entrada configurable de 0 V hasta un voltaje de referencia determinado (Vref). De establecer Vref en 5 V, se genera una diferencia en el rango de entrada del convertidor, lo que reduce la eficiencia del sistema. Para optimizar la interfaz, se alimenta con 3.3 V y se conecta el pin de referencia Vref del MCP3008 a la fuente de 3.3 V de la Raspberry Pi. De esta manera, el rango de entrada del ADC se ajusta de 0 V a 3.3 V, y así, se garantiza una digitalización completa y eficiente de la señal de pulso cardíaco. En la Ilustración 4.19. se muestran los pines del convertidor ADC MCP3008 para verificar las conexiones que se deben realizar para unir el filtro y el convertidor.

Ilustración 4.19. Conexiones del convertidor ADC MCP3008



NOTA: Las conexiones se realizan con los pines GPIO de la Raspberry Pi.

Elaboración propia.

A continuación, se detallan las conexiones de cada uno de los pines:

- CH0: Entrada de la señal que viene desde la salida del filtro.
- VREF: Voltaje de referencia conectado a los 3.3 V de la Raspberry Pi lo que define el voltaje máximo del convertidor.
- VDD: Voltaje de alimentación del convertidor. Se conecta a los 5 V de la Raspberry Pi.
- AGND y DGND: Se conecta al pin de GND de la Raspberry Pi.
- CLK: Señal de reloj que sincroniza la comunicación entre la Raspberry Pi y el convertidor. Se conecta al pin GPIO 11.
- CS/SHDN: Activa y desactiva el convertidor. Se conecta al pin GPIO 25 en la Raspberry Pi.
- DOUT y DIN: Son los pines para establecer la comunicación digital con la Raspberry Pi. Uno recibe la instrucción desde el ordenador y el otro envía la información que se requiere. Se conectan al pin GPIO 9 y al pin GPIO 10 respectivamente.

Una vez se conecta correctamente el convertidor, funciona de la siguiente manera: el convertidor recibe la información analógica del filtro y espera a que la Raspberry Pi solicite información a través del pin DIN para mandársela a través de pin DOUT de forma digital.

Al finalizar, se tiene todo el sistema totalmente conectado con todos los requerimientos de diseño, valores de los componentes, pines a utilizar y conexiones necesarias.

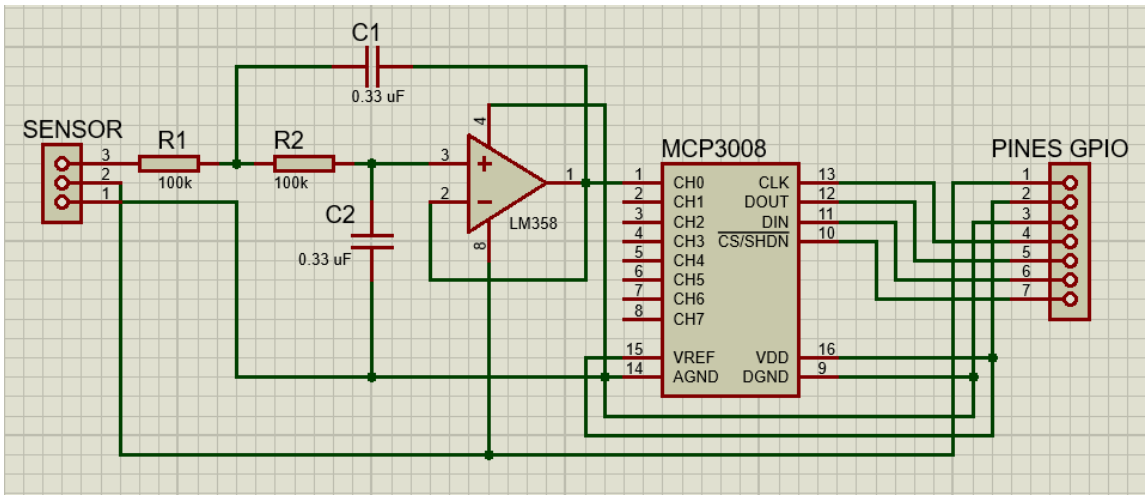
4.4.1.2. Diseño de circuitos del sistema

El diseño del asistente se desarrolla en el software Proteus. Este programa permite diseñar el esquemático, las trazas y el diseño de la placa electrónica que se implementa. Para este diseño se toman en cuenta todos los componentes anteriormente señalados.

Algunos componentes no requieren de tantas conexiones. En el caso del sensor de temperatura, se conecta directamente a los pines GPIO de la Raspberry Pi. Del mismo modo, la pantalla táctil se conecta mediante cable HDMI y USB por lo que no tiene algún otro requerimiento.

Por otro lado, el sensor de pulso si requiere, por lo que se procede con el diseño de una placa electrónica. Un inconveniente que se presenta al momento de hacer la conexión es como conectar los cables GPIO de la Raspberry Pi con los demás componentes. Para solucionar este problema, se opta por hacer uso de cables jumpers tipo macho-hembra. Una parte del cable se conecta a los pines GPIO y la otra parte se conecta a la placa mediante pines individuales. Con estos pines individuales se establece las entradas, las salidas, la alimentación y el GND de la placa electrónica. En la Ilustración 4.20. se detalla el circuito que se implementa para el sensor de pulso cardiaco.

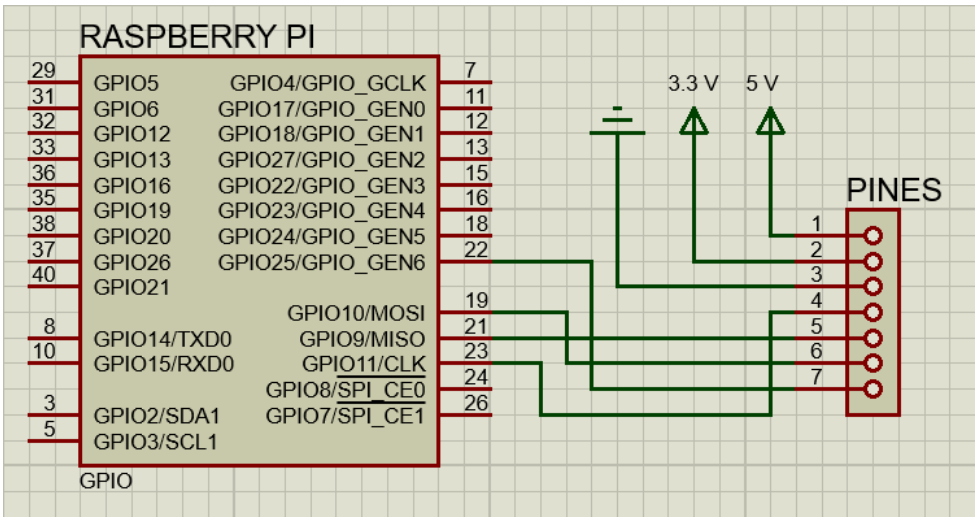
Ilustración 4.20. Diseño del circuito del sistema



NOTA: En el esquema se detallan los pines de entrada para el sensor, el filtro, el convertidor y los pines de salida GPIO.

En la imagen se aprecia la conexión de los pines GPIO con el resto de los componentes para integrar todo el sistema. En la Ilustración 4.21. se muestra como correspondería la conexión de cada uno de esos pines en la placa de la Raspberry Pi.

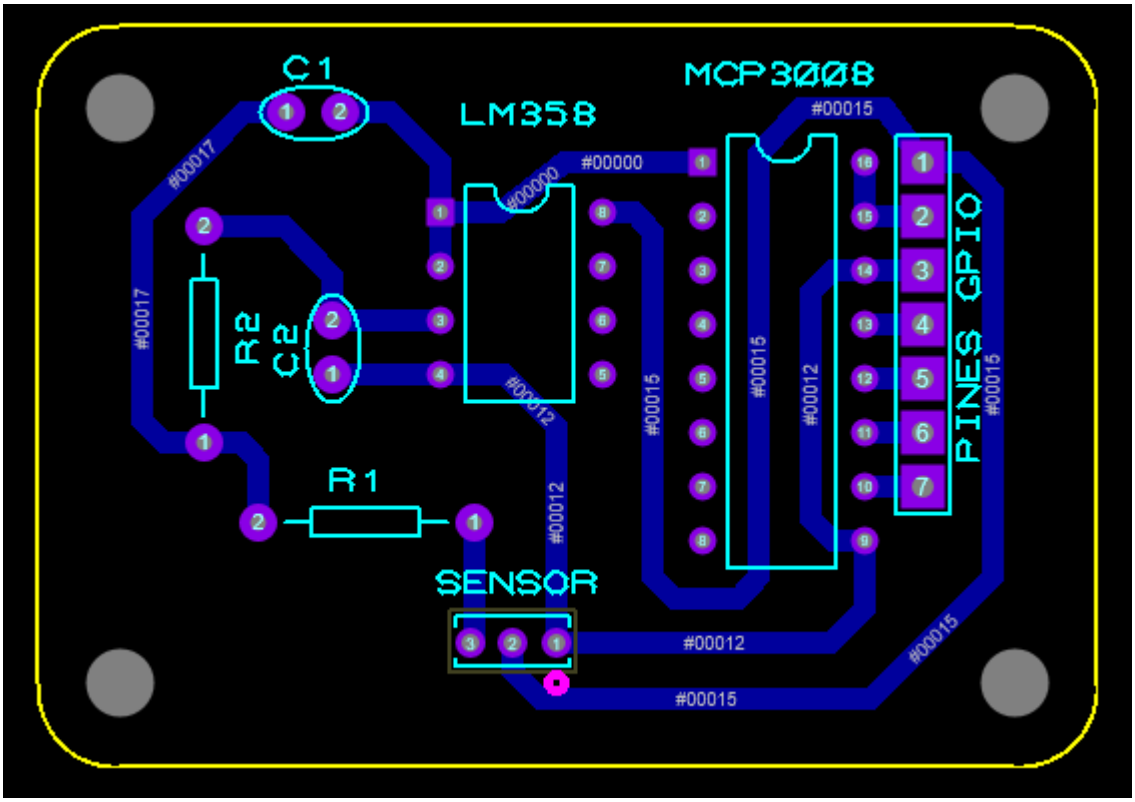
Ilustración 4.21. Conexión de los pines GPIO en la Raspberry Pi



NOTA: Los pines de alimentación también se conectan desde la Raspberry Pi, en la figura se sobre entiende.

Ahora se procede con el diseño de PCB, se acomodan los componentes de tal forma que no interfieran con las trazas que se dibujan para conectarlos entre sí. En la Ilustración 4.22. se detalla el armado y la distribución aplicada a la placa electrónica.

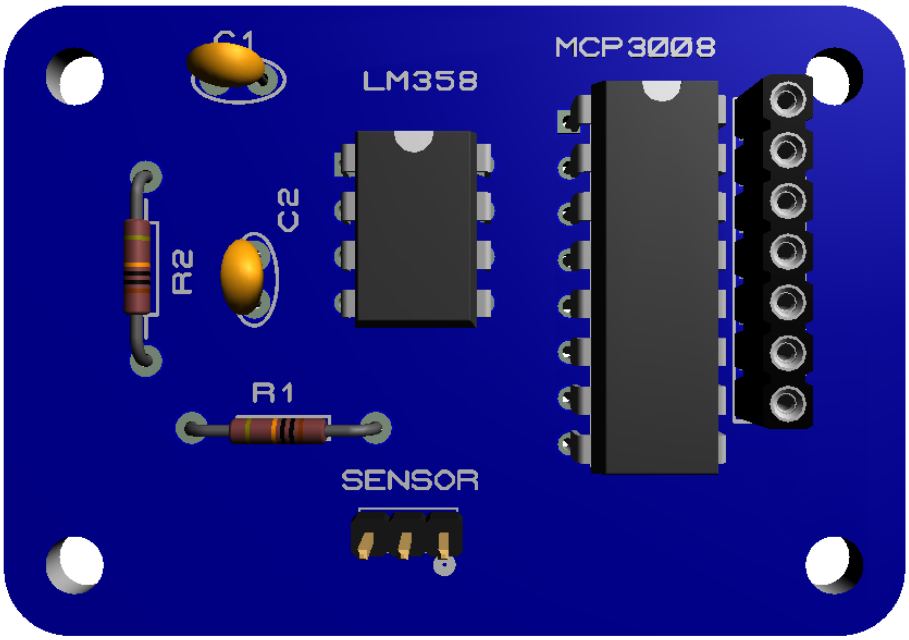
Ilustración 4.22. Diseño de PCB



NOTA: En color azul se detallan las trazas que corresponden a las conexiones entre los componentes.

Para esta ocasión, se decide usar trazas con un tamaño de T40 ya que se adecuan de mejor manera a los pines de los componentes que se utilizan en la placa. Una vez se tiene desarrollada y establecida la distribución en la placa electrónica, se realiza una simulación en 3D para tener una idea del resultado final de la placa electrónica. En la Ilustración 4.23. se detalla el diseño en 3D de la placa electrónica.

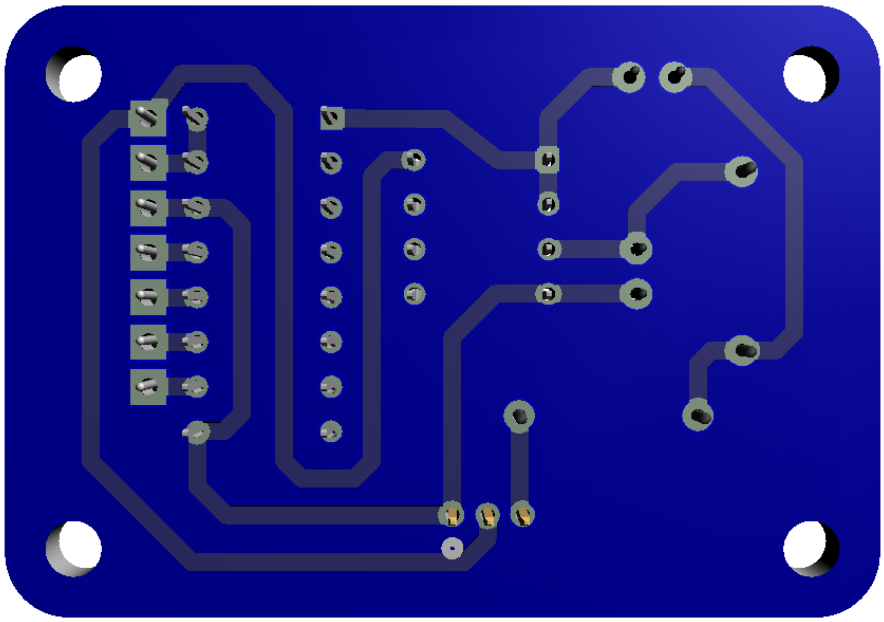
Ilustración 4.23. Diseño 3D de la placa electrónica



NOTA: El Diseño 3D simula la disposición física de los componentes considerando su tamaño real.

En la Ilustración 4.24. también se detallan las trazas diseñadas para la placa.

Ilustración 4.24. Trazas de la placa electrónica



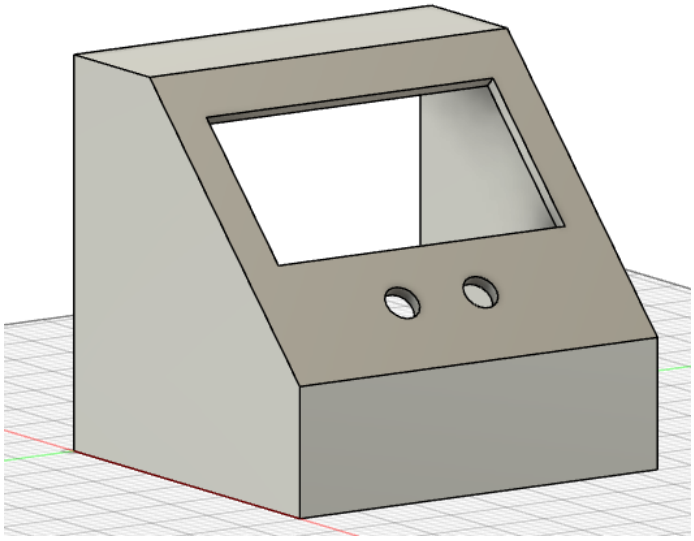
NOTA: Las trazas también corresponden al diseño 3D.

Una vez se tiene el diseño de la placa totalmente concluida, se exporta y se realiza la placa electrónica de forma física para luego realizar pruebas antes de su implementación final en el sistema terminado.

4.4.1.3. Diseño 3D del asistente

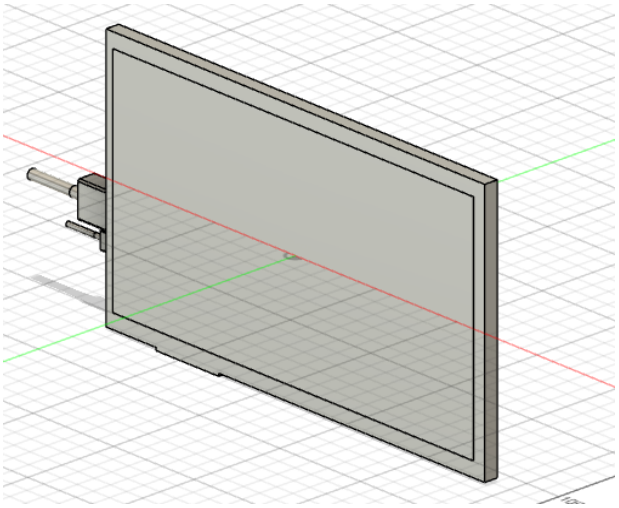
Para la estructura física del asistente se opta por realizar una impresión 3D de un diseño propio que se realiza en el software Autodesk Fusion 360. El diseño del asistente primero se realiza de forma virtual antes de su posterior impresión e implementación. Una vez se tienen todos los componentes establecidos y ordenados, se debe diseñar la estructura que va a almacenar y soportar todos los materiales del asistente. En la Ilustración 4.25. se muestra el boceto de la estructura del asistente.

Ilustración 4.25. Estructura del asistente



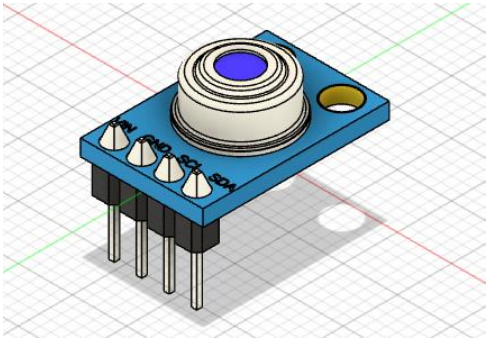
Se deja espacio para los principales dispositivos de entrada del asistente, un espacio para la pantalla y dos huecos para los sensores. Se toman en cuenta las medidas de los componentes respectivos, para eso se diseñan los mismos en 3D. En la Ilustración 4.26. se detalla la pantalla con sus dimensiones exactas.

Ilustración 4.26. Diseño 3D de la pantalla



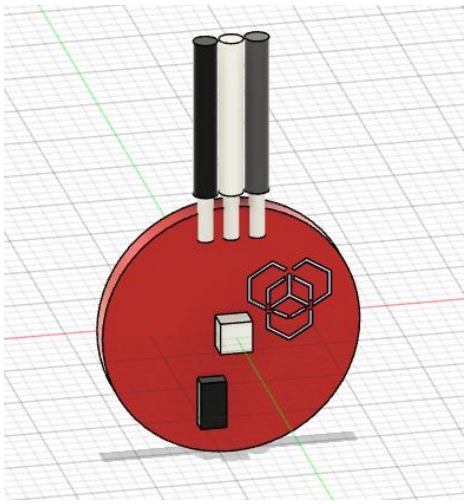
En la Ilustración 4.27. se muestra el diseño del sensor de temperatura.

Ilustración 4.27. Diseño 3D del sensor de temperatura



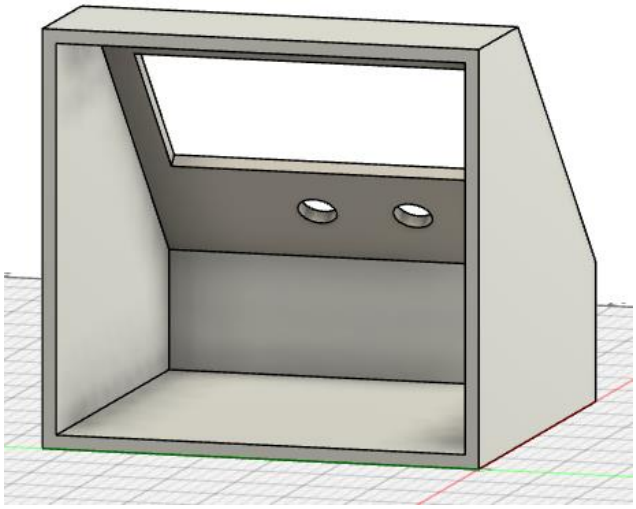
Por último, en la Ilustración 4.28. se detalla el sensor de pulso.

Ilustración 4.28. Diseño 3D del sensor de pulso



En la parte posterior se deja un especio para añadir los circuitos correspondientes y la Raspberry Pi. En la Ilustración 4.29, se detalla la parte trasera de la estructura del asistente.

Ilustración 4.29. Parte trasera del asistente



Por último, se añade una tapa para evitar que los componentes se salgan y así dejar totalmente sellado todo el sistema. Como detalle extra, se deja espacio para el conector de alimentación y los puertos de la Raspberry Pi en caso de ser requeridos.

4.4.2. Software

En la parte de software se detallan los procedimientos implementados para desarrollar la interfaz gráfica del asistente. Esta parte es muy importante ya que es la principal interacción con el usuario.

4.4.2.1. Diseño de la interfaz gráfica

El software que se implementa en el asistente se desarrolla con base a una arquitectura modular, lo que permite la integración de distintas funciones dentro de un mismo entorno y donde cada módulo tiene responsabilidades

específicas para que todo el sistema trabaje correctamente. En la Tabla 4.1. se detallan los módulos que componen el asistente.

Tabla 4.1. Módulos de software del asistente.

Módulo de adquisición de datos	Se encarga de capturar las mediciones de los sensores del asistente.
Módulo de interfaz gráfica	Permite la interacción entre el paciente y el sistema a través de la pantalla táctil.
Módulo de inteligencia artificial	Proceso los datos registrados y realiza la comunicación con el modelo de IA.

4.4.2.2. Requerimientos de la interfaz

Para el diseño del software y la interfaz del asistente se establecieron una serie de requerimientos orientados a garantizar la facilidad de uso, la eficiencia operativa y la seguridad de la información. A continuación, se detallan algunos de los requerimientos.

- Permitir el ingreso de síntomas mediante la pantalla táctil, así como algún antecedente médico, alergia a algún medicamento o intervención quirúrgica previamente realizada.
- Mostrar los resultados de manera clara y comprensible.
- Comunicar los datos al modelo de inteligencia artificial en tiempo real y con una conexión estable a la red.
- Desplegar recomendaciones preliminares comprensibles y accesibles para el paciente.
- Interactuar con facilidad con los pacientes sin importar la edad o conocimiento ya sea en el tema de la medicina o tecnología.

Por otro lado, también se establecen otros requerimientos de diseño un poco más detallados, como ser:

- Interfaz intuitiva
- Respuesta inferior a 10 segundos
- Utilizar colores suaves
- Tipografía de buena legibilidad

4.4.2.3. Desarrollo de la interfaz

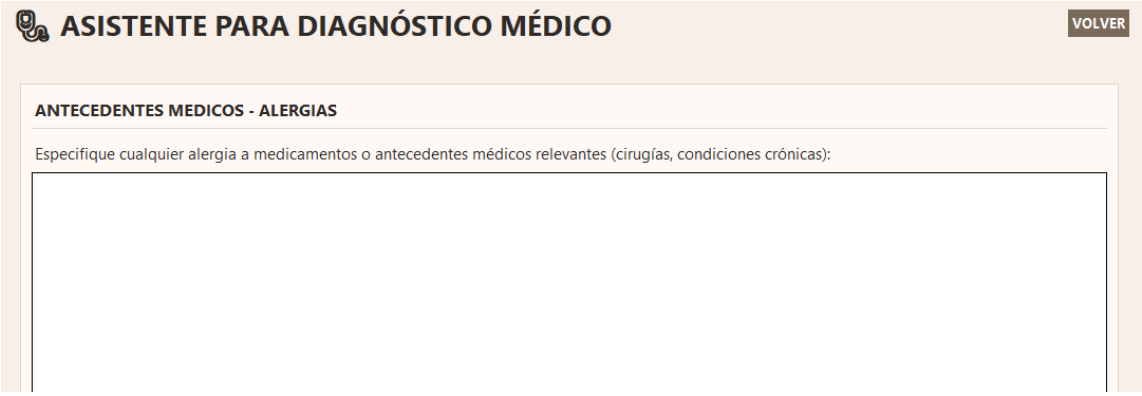
Para el desarrollo del código, se utiliza Python. Se opta por este lenguaje debido a su compatibilidad con Raspberry Pi y por la disponibilidad de librerías que facilitan tanto el manejo de hardware como el desarrollo de interfaces gráficas e inteligencia artificial. El desarrollo de la interfaz se lleva a cabo en la terminal del sistema operativo de Raspberry Pi OS Lite. Se instala un entorno gráfico para no sobrecargar todo el sistema.

Primero, se realiza el módulo de registro de los datos, para esta parte se opta por utilizar un sistema de selección múltiple, y se añade una ventana extra para anotar un síntoma que no se especifique en la lista. En la Ilustración 4.30. se muestra un boceto de la idea.

Ilustración 4.30. Boceto para la adquisición de síntomas

En adicción a los síntomas, el paciente también debe señalar si presenta alguna alergia a algún medicamento o si tiene algún antecedente médico relevante. En la Ilustración 4.31. se muestra un boceto de esta idea.

Ilustración 4.31. Boceto para la adquisición de datos extra del paciente



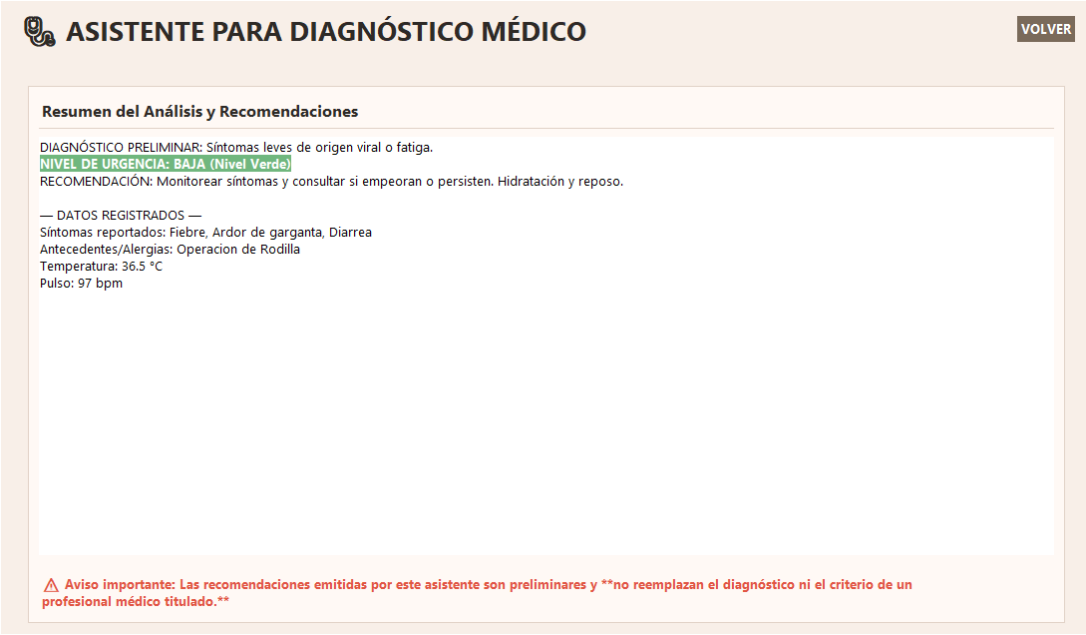
Por último, se añade una parte para realizar la medición de los datos fisiológicos del paciente. En la Ilustración 3.32. se muestra el boceto para medir la temperatura y el pulso.

Ilustración 4.32. Boceto para la medición de temperatura y pulso cardiaco

Una vez se registran los síntomas, se pasa al módulo de inteligencia artificial. En esta etapa, el sistema con toda la información del paciente llama a la API de la inteligencia artificial para establecer una comunicación mediante la red y así obtener la respuesta requerida. La inteligencia artificial analiza todos los síntomas y datos entregados por el paciente para generar un veredicto respecto al caso.

Luego de recibir la respuesta de la inteligencia artificial, en el módulo de interfaz gráfica, se muestran los resultados generados. En la Ilustración 4.33. se muestra la interfaz gráfica mostrando los resultados.

Ilustración 4.33. Boceto donde se muestra el resultado del análisis



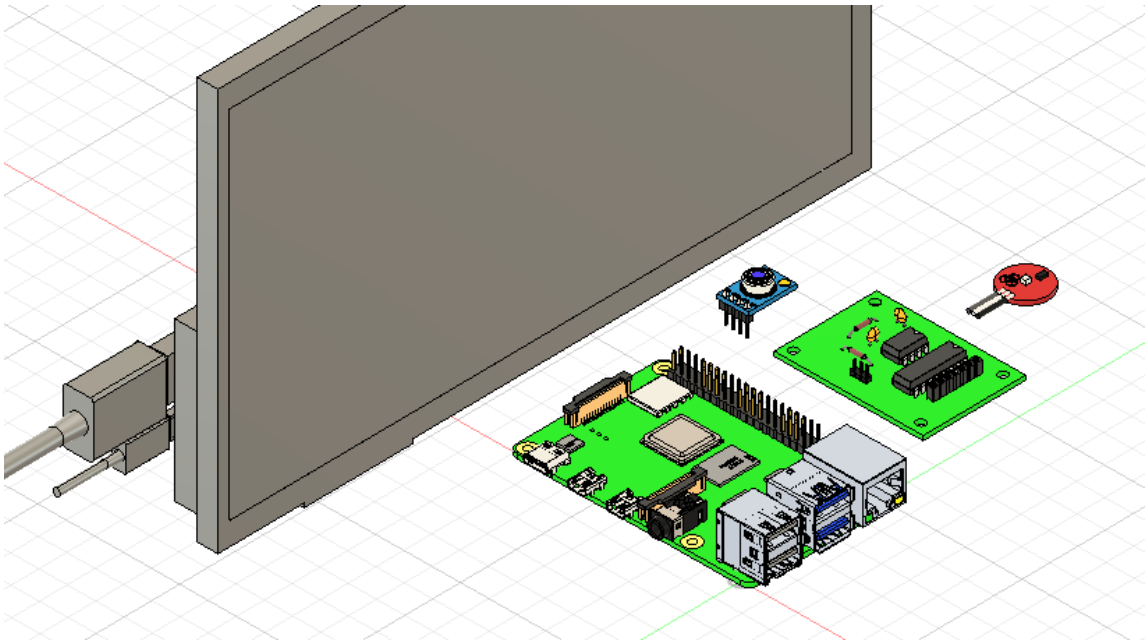
NOTA: Los valores de los sensores, los antecedentes y los síntomas no son reales, se establecen para verificar el funcionamiento del prototipo.

En esta parte también se hace un pequeño resumen de la situación del paciente antes de dar la respuesta. Se determina que tan grave es su caso y aproximadamente cuanto va a esperar para ser atendido. Luego, se dan algunas recomendaciones preliminares para aliviar los malestares. Por último, se recalca que el resultado no reemplaza el diagnóstico de un profesional de salud.

4.4.3. Sistema en conjunto

En esta sección, se muestra un modelado de los componentes que integran el asistente con todas las partes totalmente implementadas. En la Ilustración 4.34. se muestra una modelo de todo el sistema y sus componentes.

Ilustración 4.34. Simulación de los componentes del sistema



NOTA: En la figura se ven todos los componentes que se utilizan, aun sin las conexiones. Elaboración propia.

Una vez se tienen todos los componentes calculados, diseñados y simulados, se procede con la implementación de todo el sistema.

CAPÍTULO 5

IMPLEMENTACIÓN DEL ASISTENTE PARA DIAGNÓSTICO MÉDICO

5. IMPLEMENTACIÓN DEL ASISTENTE PARA DIAGNÓSTICO MÉDICO

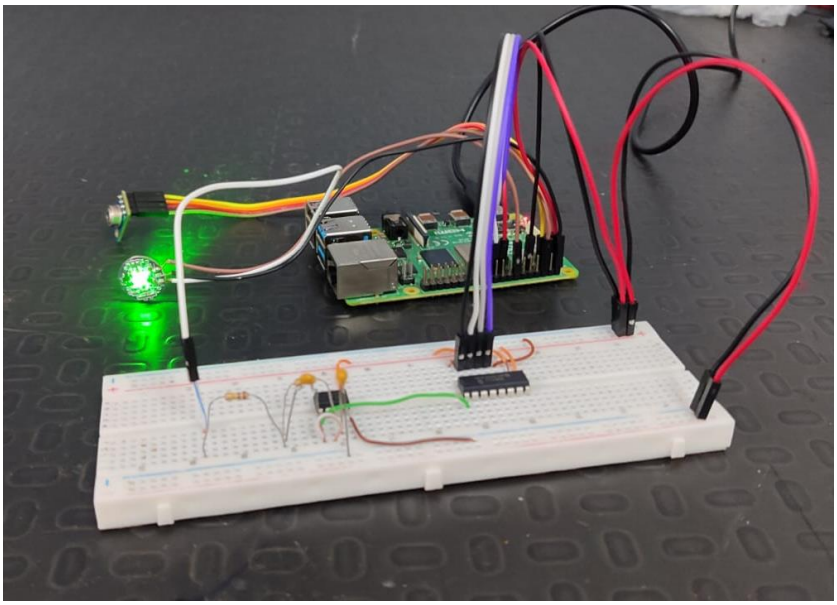
5.1. Montaje del prototipo

En esta sección, se muestra la fabricación de los circuitos y conexiones necesarias para la integración del asistente para diagnóstico médico.

5.1.1. Fabricación de la placa electrónica

Antes del desarrollo de la placa electrónica de forma final, primero se realizan pruebas en una placa protoboard para verificar que las conexiones estén correctamente realizadas y que no existan posibles equivocaciones o sobrecalentamiento de los componentes. En la Ilustración 5.1. se muestra la placa protoboard armada y conectada.

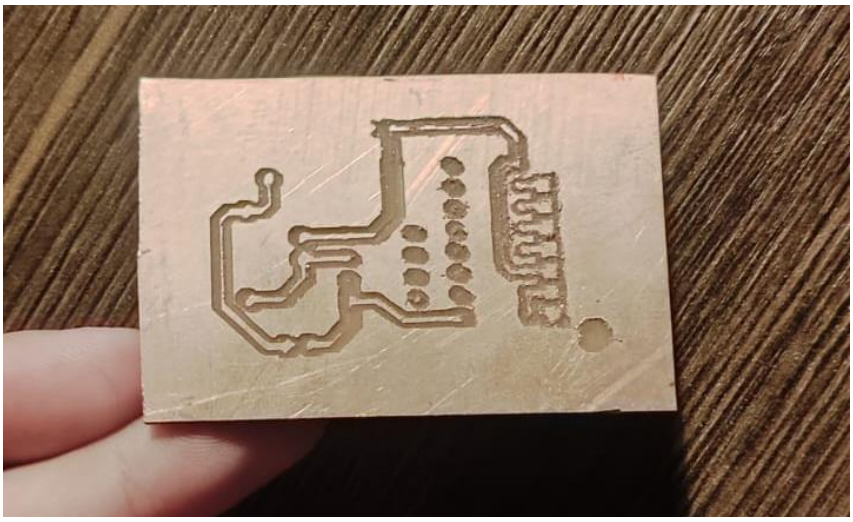
Ilustración 5.1. Circuito en Protoboard



Una vez se confirman y se verifican las conexiones, mediante el software Proteus se exporta el circuito para imprimir y pasar a una placa de cobre. Se coloca la impresión del circuito sobre la placa de cobre y se aplica calor para pasar las trazas del papel a la placa. Este proceso lleva tiempo y se hace con

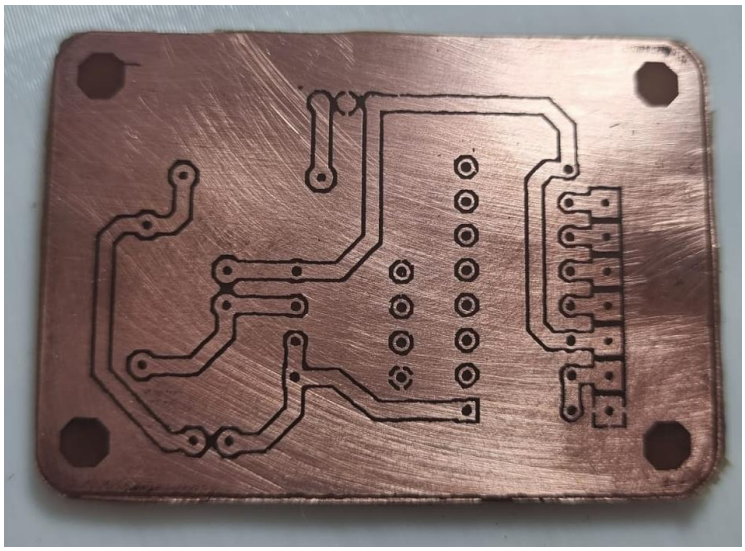
cuidado ya que, si las trazas salen mal, la placa no funcionará correctamente. En la Ilustración 5.2. se muestran las trazas en la placa de cobre.

Ilustración 5.2. Trazas sobre la placa de cobre



Luego de realizar las trazas, se realizan las perforaciones en la placa en los lugares correspondientes para poder añadir los componentes. En la Ilustración 5.3. se muestran las perforaciones en la placa.

Ilustración 5.3. Placa perforada

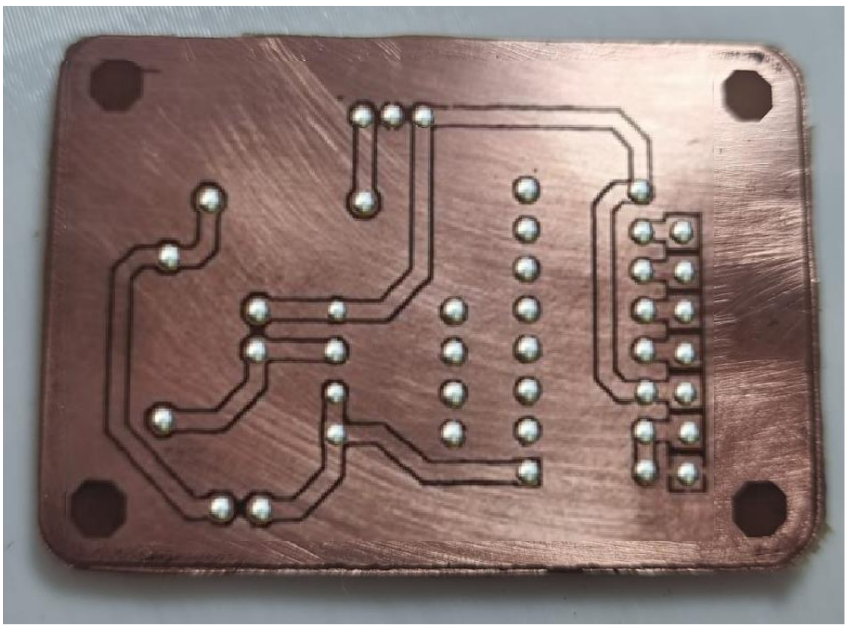


Con las perforaciones hechas, se colocan todos los componentes donde corresponden y se realiza la soldadura a cada uno de los pines. En la Ilustración 5.4. se pueden ver los componentes colocados en la placa.

Ilustración 5.4. Componentes en la placa

En la Ilustración 5.5. se muestra el reverso de la placa y se muestran las soldaduras de los componentes.

Ilustración 5.5. Soldadura de los componentes

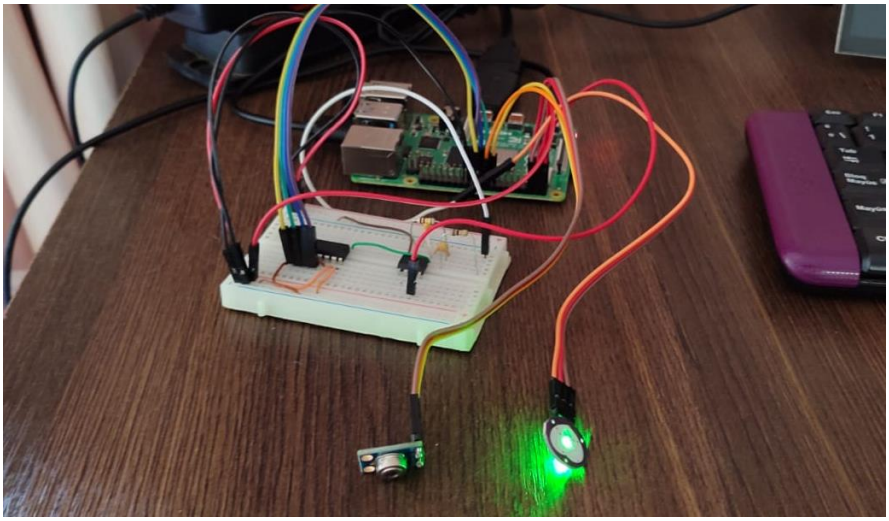


De esta forma, se finaliza la fabricación de la placa y se conecta con el resto de los componentes.

5.1.2. Conexión de los componentes

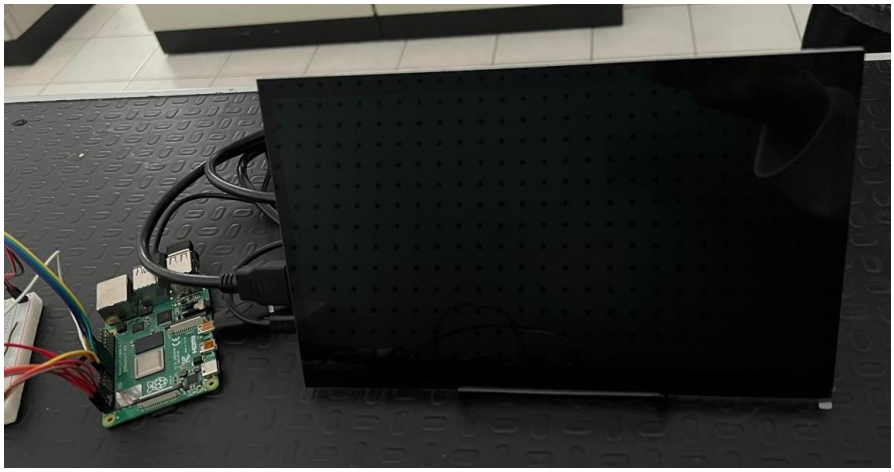
Para la conexión de los componentes, se toman en cuentas todos los componentes y dispositivos que conforman el asistente. En la Ilustración 5.6. se muestran las conexiones de los sensores y el circuito con la Raspberry Pi.

Ilustración 5.6. Conexión de los sensores



En la Ilustración 5.7. se muestra la conexión de la pantalla táctil con la Raspberry Pi.

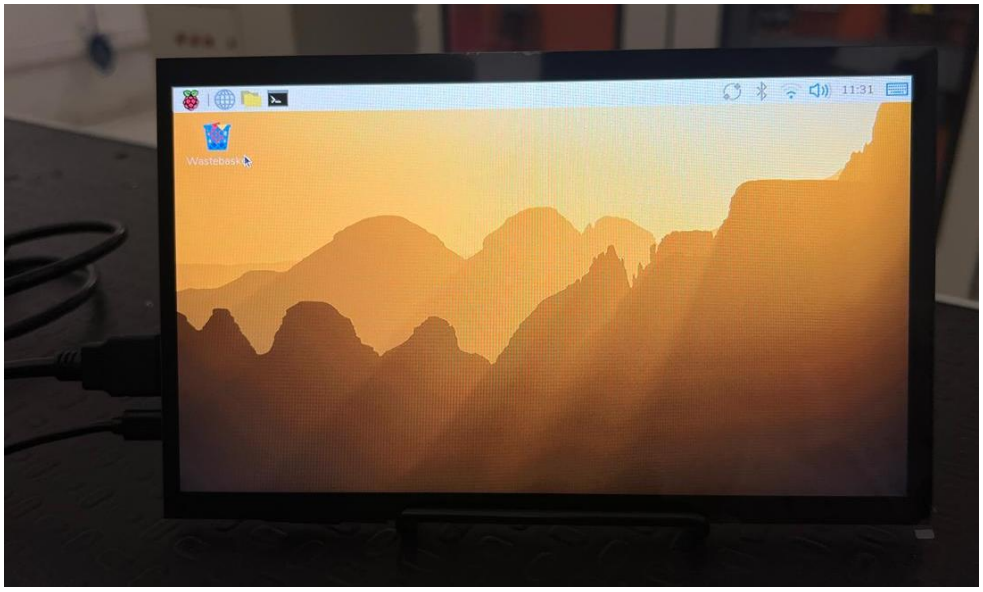
Ilustración 5.7. Conexión de pantalla táctil con Raspberry Pi



5.1.3. Verificación de funcionamiento

En esta sección se comprueba el correcto funcionamiento del asistente solo con las conexiones básicas, es decir, solo con alimentación, sin tomar en cuenta la programación en el computador o la adquisición de datos. En la Ilustración 5.8. se muestran las conexiones funcionando correctamente.

Ilustración 5.8. Verificación de funcionamiento de conexiones



5.2. Conectividad y almacenamiento

5.2.1. Configuración para comunicación entre interfaces

La conectividad entre los componentes del asistente se establece mediante interfaces de comunicación digital estándar, garantizando un flujo de datos estable entre los sensores, el convertidor y la Raspberry Pi.

El sensor de temperatura MLX90614 se comunica con la Raspberry Pi a través del protocolo I2C, utilizando los pines SDA y SCL (GPIO 2 y GPIO 3) para el intercambio de información en tiempo real [44]. Por otro lado, el convertidor analógico–digital MCP3008 emplea el protocolo SPI para

transmitir los datos provenientes del sensor de pulso cardíaco COMIMARK, utiliza los pines MOSI, MISO, CLK y CS de la Raspberry Pi [42].

Ambos protocolos se habilitan directamente desde la configuración del sistema operativo. Esto permite que sean usados al momento de realizar la programación y poder tomas mediciones.

5.2.2. Integración con inteligencia artificial

Para la aplicación de la inteligencia artificial se opta por utilizar la inteligencia artificial de Gemini 2.5 Flash. Se escoge esta opción debido a su fácil acceso y que no requiere ningún costo adicional. Es una inteligencia artificial externa por lo que se realiza mediante conexión de red. Presenta algunas restricciones al momento de su implementación, como intentos limitados o cierto número de tokens permitidos. Su implementación en la programación se da mediante una clave de API. Esta clave se utiliza para llamar a la inteligencia artificial cada que se la necesite a lo largo del código. La clave API llama a Gemini y procesa toda la información del paciente.

5.2.3. Implementación de base de datos

Para el almacenamiento, se decide usar SQ Lite como sistema de gestión de base de datos del prototipo. Se escoge esta opción ya que se integra de manera nativa, fácil y gratuita. Ayuda a eliminar la complejidad de instalar y programar un servidor de base de datos tradicional y optimiza el rendimiento del sistema.

Se implementa con el objetivo de tener un historial de los pacientes que hacen uso del prototipo. De la misma forma, ayuda a visualizar de mejor manera las lecturas de los sensores, los síntomas y la clasificación de gravedad.

Adicionalmente, se implementa un botón para ordenar los pacientes del más grave al menos grave y otro para eliminar el registro que se requiera.

5.3. Desarrollo del software

5.3.1. Estructura general del sistema

El software del asistente se desarrolla íntegramente en Python, ya que así se aprovecha su compatibilidad con la plataforma Raspberry Pi 4. Se desarrollan los 3 módulos que componen el sistema completo del asistente.

El primer módulo se enfoca en la adquisición de datos. Aquí se diseña y programa el registro de información del paciente. Los datos se registran mediante los sensores y los síntomas, antecedentes o alergias mediante la pantalla táctil.

El segundo módulo consiste en el diseño de la interfaz gráfica. Para esto se opta por usar un diseño moderno y a la vez profesional, que sea amigable con el paciente y que sea fácil de usar. La interfaz gráfica se desarrolla en su totalidad con Tkinter.

Por último, el tercer módulo consiste en la aplicación de la inteligencia artificial al asistente. Se aplica mediante la API de Gemini 2.5 Flash. Se crea un contexto para que Gemini interprete toda la información con los síntomas, datos y antecedentes del paciente y así entregar los resultados generados.

5.3.2. Desarrollo del software

En esta sección se detallan partes importantes del código para ayudar a entender cómo funciona toda su estructura. Se describe la implementación de los sensores de temperatura, de pulso y la conectividad con la inteligencia artificial.

5.3.2.1. Lectura del sensor de temperatura

Para la implementación del sensor MLX90614 se requiere instalar algunas librerías para poder desarrollar la función que se encarga de leer los valores. En la Tabla 5.1. se muestran las librerías necesarias.

Tabla 5.1. Tabla de librerías para el sensor MLX90614

LIBRERIAS	USO EN EL CODIGO
smbus2	Se encarga de la comunicación I2C
time	Controla el tiempo de transmisión

En la Ilustración 5.9. se muestra el código en la programación que se usa para leer los valores del sensor.

Ilustración 5.9. Código para lectura del sensor MLX90614

```python
# --- FUNCIÓN DE LECTURA I2C ---
def read_temp_data(register):
    """
    Lee un valor de 16 bits del sensor MLX90614 y lo convierte a grados Celsius.
    """
    try:
        # Crea una instancia del bus I2C
        bus = smbus2.SMBus(BUS_NUM)

        # Lee 2 bytes (16 bits) del registro especificado
        data = bus.read_word_data(MLX_ADDR, register)

        # Cierra el bus (buena práctica)
        bus.close()

        # El sensor almacena la temperatura en Kelvin * 0.02.
        # Conversión: (Valor_Crudo * 0.02) - 273.15 = Celsius
        celsius = (data * 0.02) - 273.15

        return celsius
    except Exception as e:
        # Imprime el error y asegura el cierre de recursos
        print(f"\nError de comunicación I2C: {e}")
        print("Verifique que el sensor esté conectado a GPIO 2 (SDA) y GPIO 3 (SCL).")
        return None
```

5.3.2.2. Lectura del sensor de pulso

Para la lectura del valor obtenido del sensor de pulso, es necesaria la implementación de un algoritmo de pulso. El sensor pasa por el filtro y luego por el convertidor MCP3008, pero su salida equivale a una serie de valores

de voltaje, es decir, la señal de salida no está en BPM. BPM es la unidad de medida que determina la cantidad de latidos que una persona tiene en un minuto. El sistema procesa la información en BPM por lo que se usa un algoritmo para la conversión de voltaje a dicha unidad. En la Tabla 5.2. se muestran las librerías necesarias en Python para el algoritmo de conversión.

Tabla 5.2. Tabla de librerías para el sensor de pulso

LIBERIAS	USO EN EL CODIGO
board	Proporciona acceso a los pines GPIO de Raspberry Pi
busio	Se encarga de la comunicación entre el hardware del sistema
digitalio	Sirve para manejar y configurar los pines digitales
Adafruit_mcp3008	Sirve para la comunicación con el convertidor MCP3008

En la Ilustración 5.10. se muestra la implementación de un código en la programación para realizar la conversión y la lectura del sensor.

Ilustración 5.10. Algoritmo de BPM para el sensor de pulso

```
# --- Lógica de Detección de Pico ---

# 1. El valor cruzó el umbral HACIA ARRIBA
if valor_bruto > THRESHOLD and not beat_detected:
    beat_detected = True # Marcamos que estamos "dentro" de un pico

    # Solo calculamos IBI si NO es el primer latido
    if last_beat_time != 0:
        ibi = current_time - last_beat_time

        # Filtro de sanidad: un pulso humano está entre 0.25s (240 BPM) y 1.5s (40 BPM)
        if 0.25 < ibi < 1.5:
            ibi_list.append(ibi)

            # Mantenemos solo los últimos N latidos en la lista
            if len(ibi_list) > NUM_BEATS_TO_AVERAGE:
                ibi_list.pop(0) # Elimina el más antiguo

            # Se calcula el BPM promedio
            if ibi_list:
                avg_ibi = sum(ibi_list) / len(ibi_list)
                current_bpm = 60 / avg_ibi

    last_beat_time = current_time # Guardamos la hora de este pico

# 2. El valor volvió a bajar (salió del pico)
elif valor_bruto < THRESHOLD and beat_detected:
    beat_detected = False # Estamos listos para buscar el próximo pico
```

5.3.2.3. Conexión con la inteligencia artificial

Para conectar el sistema con IA se debe importar la librería "google.generativeai". Se instala la librería y se añade el código para llamar a la inteligencia artificial. Se añade un "prompt" para darle instrucciones extra a Gemini y así realice un mejor análisis. En la Ilustración 5.11. se muestra su desarrollo en la programación.

Ilustración 5.11. Implementación de IA en la programación

```
    def perform_ai_analysis(self, symptoms, history, temp, pulse):
        """Realizar el análisis con IA (ejecutado en thread separado)"""
        all_symp = [s if not s.startswith("Otros:") else s[7:].strip() for s in symptoms]
        symptoms_text = ', '.join(all_symp) if all_symp else 'Ninguno reportado'

        prompt = f"""Eres un asistente médico experto. Analiza cuidadosamente la siguiente información del paciente.

DATOS DEL PACIENTE:
- Síntomas: {symptoms_text}
- Antecedentes médicos: {history if history else 'No especificados'}
- Temperatura corporal: {temp if temp != 0.0 else 'No medida'} °C
- Frecuencia cardíaca: {pulse if pulse != 0 else 'No medida'} bpm
```

También se añaden algunas instrucciones de redacción que debe seguir la IA para obtener una respuesta ordenada y completa. En la Ilustración 5.12. se muestra el tipo de respuesta que se espera.

Ilustración 5.12. Instrucciones de redacción para Gemini

```
INSTRUCCIONES CRÍTICAS:
Debes responder EXACTAMENTE en este formato, línea por línea:

DIAGNÓSTICO: [escribe aquí un diagnóstico preliminar específico en UNA línea]

NIVEL_URGENCIA: [escribe EXACTAMENTE una de estas tres opciones sin agregar nada más:
ALTA URGENCIA (Nivel Rojo)
MODERADA (Nivel Amarillo)
BAJA (Nivel Verde)]

RECOMENDACIÓN: [escribe aquí una recomendación médica específica en UNA línea]

DETALLES: [escribe aquí una explicación médica detallada de 3-5 líneas que justifique tu diagnóstico
y nivel de urgencia basándote en los signos vitales y síntomas presentados]

CRITERIOS IMPORTANTES:
- Fiebre >39°C = considerar alta urgencia
- Taquicardia >110 bpm o bradicardia <50 bpm = considerar alta urgencia
- Múltiples síntomas severos combinados = aumentar nivel de urgencia
- Antecedentes médicos relevantes = considerar en el análisis
- Este es un diagnóstico preliminar, SIEMPRE recomendar valoración médica profesional
```

5.3.3. Verificación de funcionamiento

Para comprobar que el sistema esté funcionando correctamente, se añade información de un paciente para que haga la evaluación y así, se verifique el diagnóstico que realiza el asistente. Primero, se programa una pantalla de inicio, esto sirve como bienvenida al asistente. En esta pantalla, se muestran tres botones para añadir un nuevo registro, para ver el historial de todos los pacientes en la base de datos y para salir del programa. En la Ilustración 5.13. se muestra el diseño del menú principal con las tres opciones anteriormente mencionadas.

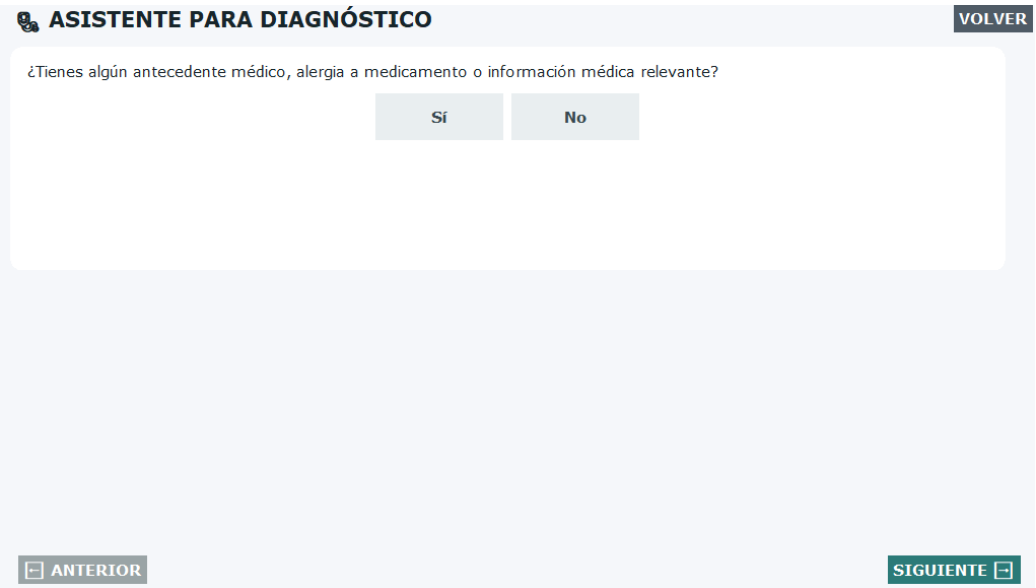
Ilustración 5.13. Diseño del menú principal

Se selecciona la opción de nuevo diagnóstico y se añade la información requerida. Los botones son de selección y también se incluye la opción de añadir otro síntoma, al seleccionar esta opción se despliega otro recuadro. Del mismo modo, se adiciona un botón para pasar a la siguiente pestaña y otro para volver al menú principal. En la Ilustración 5.14. se muestra la pestaña para registrar síntomas.

Ilustración 5.14. Pestaña para añadir síntomas

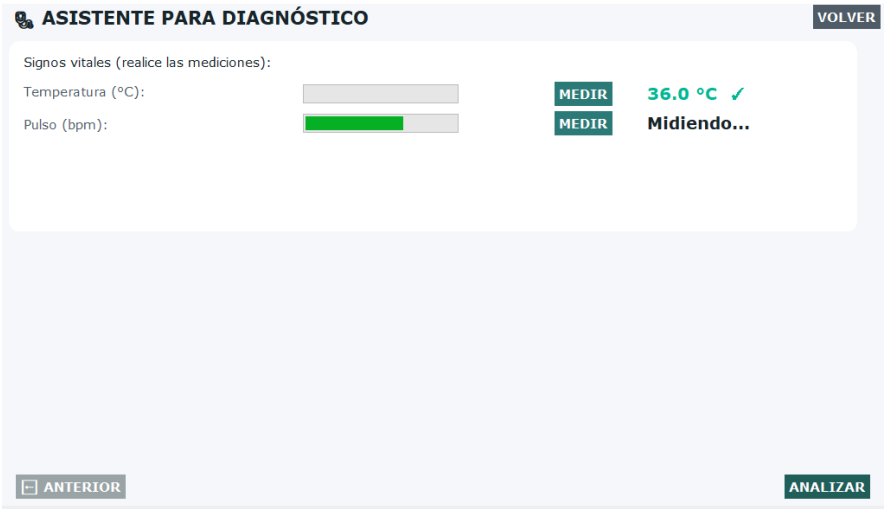
Luego se pasa a la pestaña siguiente donde se pregunta si existe algún antecedente o alergia a algún medicamento. Se selecciona la opción que corresponda y en caso de ser "Si" se despliega un recuadro para detallar la información. Además, se añade un botón para volver a la pestaña anterior. En la Ilustración 5.15. se muestra el diseño de esta pestaña.

Ilustración 5.15. Pestaña de antecedentes médicos



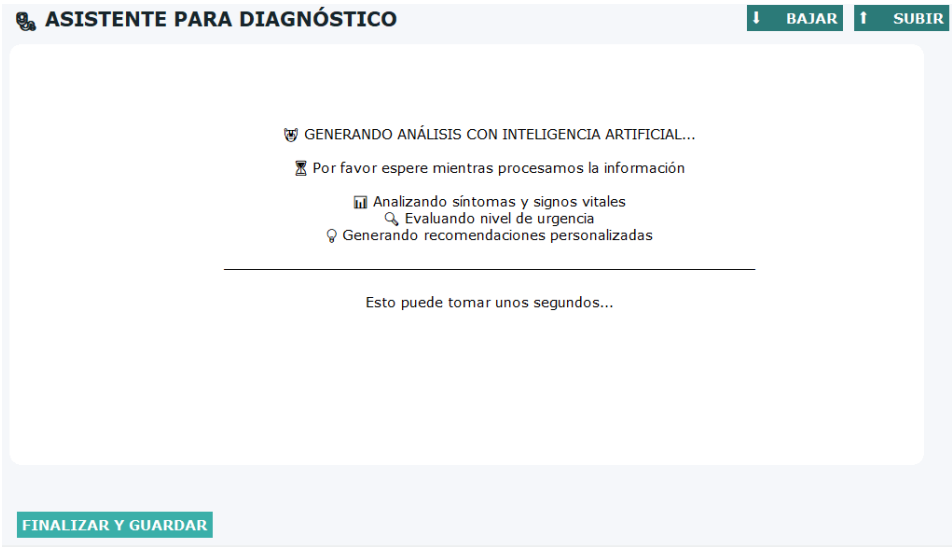
En la Ilustración 5.16. se muestra la pestaña para realizar las mediciones de los sensores con sus botones respectivos para cada acción.

Ilustración 5.16. Pestaña para medición de signos vitales



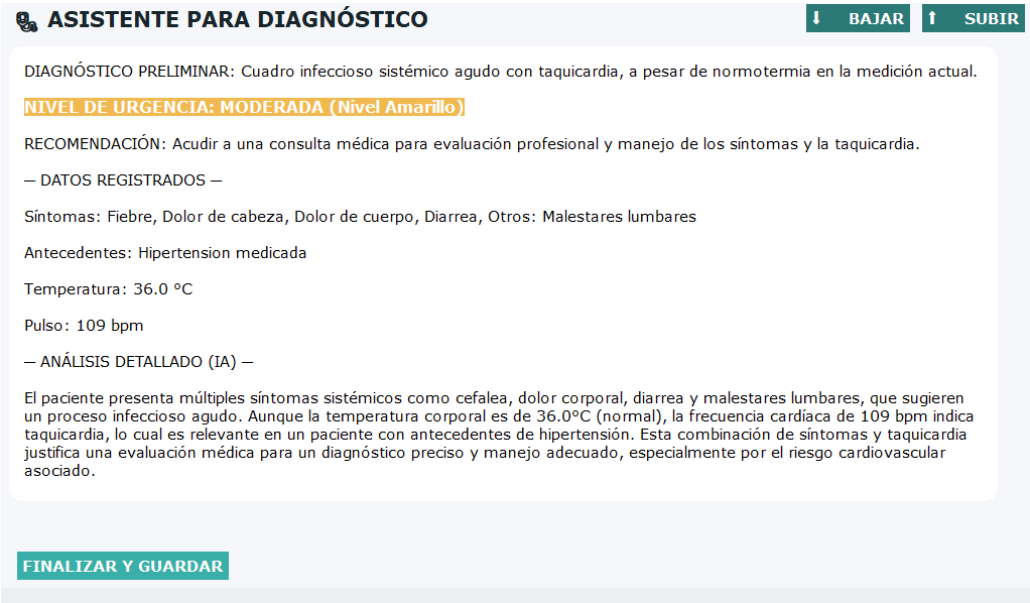
Una vez se registran toda la información, se manda para ser procesada por Gemini a través de la clave API. Esta clave se implementa en el código para ser utilizada solo al momento de analizar la situación de cada paciente. Ya que existe una demora en hacer el análisis, se opta por añadir una pestaña de espera. En la Ilustración 5.17. se muestra la pestaña mientras carga la IA.

Ilustración 5.17. Pestaña de espera



Para concluir el registro, se muestra la pestaña que enseña los resultados generados por la inteligencia artificial. Los resultados se programan para ser mostrados de forma ordenada y para que el paciente los entienda fácilmente. Se añaden botones de subir y bajar ya que la información puede ser extensa. También, se añade un botón para finalizar el registro y guardar la información. Se muestra el diagnóstico realizado por la inteligencia artificial. Por último, se subraya en color, ya sea rojo, amarillo o verde, la gravedad del caso que se presenta, siendo rojo el más urgente y verde el menos. En la Ilustración 5.18. se muestra el resultado del diseño.

Ilustración 5.18. Pestaña de resultados



Adicionalmente, si se retorna al inicio y se selecciona la opción para ver el historial, se muestra la base de datos de pacientes. Esta lista se actualiza conforme se van añadiendo pacientes nuevos al sistema. También, se implementan botones para eliminar u ordenar a los pacientes, además de un botón para volver al inicio. La información se clasifica según los datos que

se registran al momento de interactuar con el paciente. El historial se muestra en forma de tabla donde se especifican los siguientes datos:

- ID: como el número de paciente
- Fecha y hora: para contar con un registro
- Nivel de urgencia: el cual se toma como parámetro para ordenar la lista
- Temperatura: medida por su sensor
- Pulso: medida por el sensor
- Diagnostico preliminar: generado por la inteligencia artificial

En la Ilustración 5.19. se muestra la pestaña del historial de pacientes con algunos ejemplos para verificar su funcionamiento.

Ilustración 5.19. Pestaña de historial de pacientes

Id	Fecha Hora	Nivel Urgencia	Temperatura	Pulso	Diagnostico Preliminar
26	2025-10-26 17:30:12	MODERADA (Nivel Amarill	36.0	109	Cuadro infeccioso sistémico agudo con taquicardia, a pesar c
25	2025-10-26 14:23:31	MODERADA (Nivel Amarill	39.0	92	Síndrome febril agudo con síntomas sistémicos
24	2025-10-25 16:52:41	MODERADA (Nivel Amarill	38.0	118	Posible infección viral común.
23	2025-10-25 16:24:58	MODERADA (Nivel Amarill	38.2	90	Posible infección viral común.
22	2025-10-25 15:58:07	MODERADA (Nivel Amarill	38.1	56	Posible infección viral común.
21	2025-10-25 15:45:43	BAJA (Nivel Verde)	None	None	Síntomas leves de origen viral.
20	2025-10-25 15:44:40	BAJA (Nivel Verde)	36.5	55	Síntomas leves de origen viral.
19	2025-10-25 15:06:30	BAJA (Nivel Verde)	36.4	101	Síntomas leves de origen viral.
18	2025-10-25 15:04:58	MODERADA (Nivel Amarill	38.9	84	Posible infección viral común.
17	2025-10-25 15:03:49	BAJA (Nivel Verde)	39.4	81	Síntomas leves de origen viral.
16	2025-10-25 14:59:52	MODERADA (Nivel Amarill	38.9	115	Posible infección viral común.
15	2025-10-25 14:40:55	BAJA (Nivel Verde)	38.9	60	Síntomas leves de origen viral.
14	2025-10-25 13:13:57	MODERADA (Nivel Amarill	38.3	120	Posible infección viral común (gripe, resfrío) o gastrointestina
13	2025-10-25 13:00:11	MODERADA (Nivel Amarill	36.3	110	Posible infección viral común (gripe, resfrío) o gastrointestina
12	2025-10-25 12:40:29	MODERADA (Nivel Amarill	38.1	88	Posible infección viral común (gripe, resfrío) o gastrointestina
11	2025-10-25 10:39:26	BAJA (Nivel Verde)	36.6	95	Síntomas leves de origen viral o fatiga.

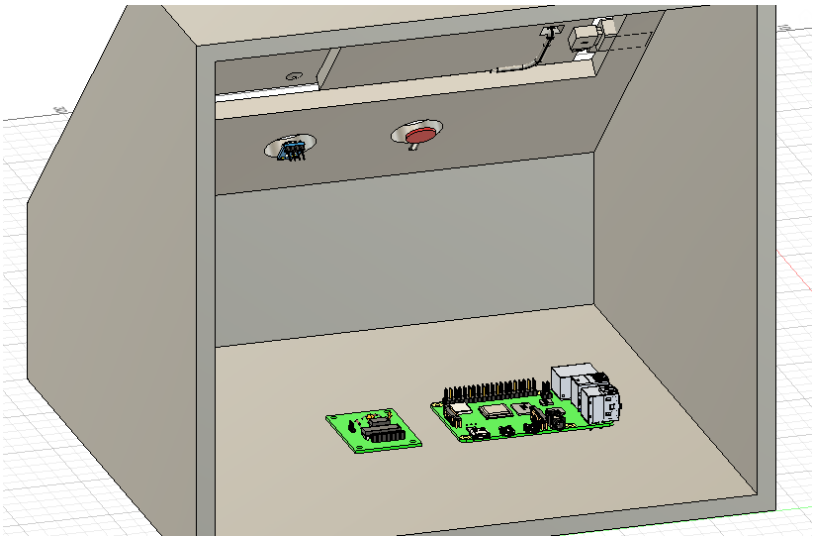
5.4. Ensamblaje del prototipo

5.4.1. Impresión de la estructura del prototipo

Una vez se tienen todos los componentes y la estructura del asistente diseñadas en 3D se procese a realizar la impresión para su posterior

aplicación. Antes de su impresión, es necesario hacer ciertos ajustes técnicos en el software Fusion Autodesk. En la Ilustración 5.20. se muestra el diseño modelado en 3D totalmente terminado y corregido.

Ilustración 5.20. Modelado en 3D completo



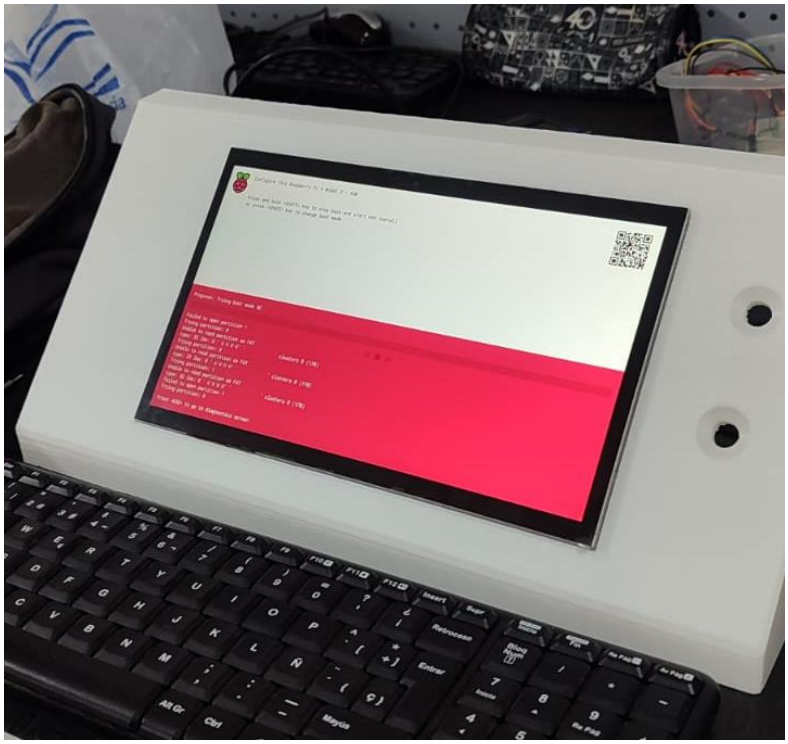
En la Ilustración 5.21. se enseña el prototipo totalmente terminado, con los ajustes requeridos y listo para su implementación.

Ilustración 5.21. Estructura impresa en 3D

5.4.2. Armado del asistente

Se procede al armado del asistente juntando todas las partes, componentes y dispositivos que conforman todo el sistema. Para eso se realizan los ajustes correspondientes a los componentes que lo requieran para así evitar algún inconveniente o impedimento al momento de su funcionamiento. Se realizan las conexiones entre componentes mediante cables jumpers, ya sea de tipo hembra o macho, según corresponda. En la Ilustración 5.22. se muestra todo el asistente terminado con todas sus partes incluidas.

Ilustración 5.22. Asistente terminado y ensamblado



5.5. Costo de implementación

La implementación del prototipo implica la adquisición de dispositivos tecnológicos y componentes electrónicos. En la Tabla 5.3. se detallan los costos de todos los materiales del asistente.

Tabla 5.3. Tabla de costos de materiales

COMPONENTE	CANTIDAD	COSTO UNITARIO (Bs)	COSTO TOTAL (Bs)
Raspberry Pi 4 Modelo B 4 GB	1	1580	1580
Pantalla HMTECH 10.1 pulgadas	1	383	383
Sensor MLX9614	1	140	140
Sensor COMIMARK	1	56	56
Convertidor ADC MCP3008	1	30	30
Resistencia100k Ohm	2	1	2
Capacitor 0.39 uF	2	3	6
Amplificador Operacional LM358	1	10	10
Impresión de la estructura en 3D	1	700	700
Varios	1	50	50
TOTAL			2957

En la Tabla 5.4. se muestran los costos de la mano de obra del prototipo.

Tabla 5.4. Costos de mano de obra del prototipo

	COSTO HORA (Bs)	HORAS TRABAJADAS	COSTO TOTAL (Bs)
Investigación	130,85	25	3271,25
Diseño electrónico	130,85	40	5234
Diseño 3D	130,85	30	3925,50
Simulación	130,85	20	2617
Programación	130,85	25	3271,25
Armado	130,85	15	1962,75
Pruebas	130,85	10	1308,50
TOTAL		165	21590,25

El valor del costo hora-hombre para la mano de obra se toma referenciado al arancel determinado por el CIEESC [50].

CAPÍTULO 6

PRUEBAS Y ANÁLISIS DE RESULTADOS

6. PRUEBAS Y ANÁLISIS DE RESULTADOS

6.1. Pruebas funcionales de hardware

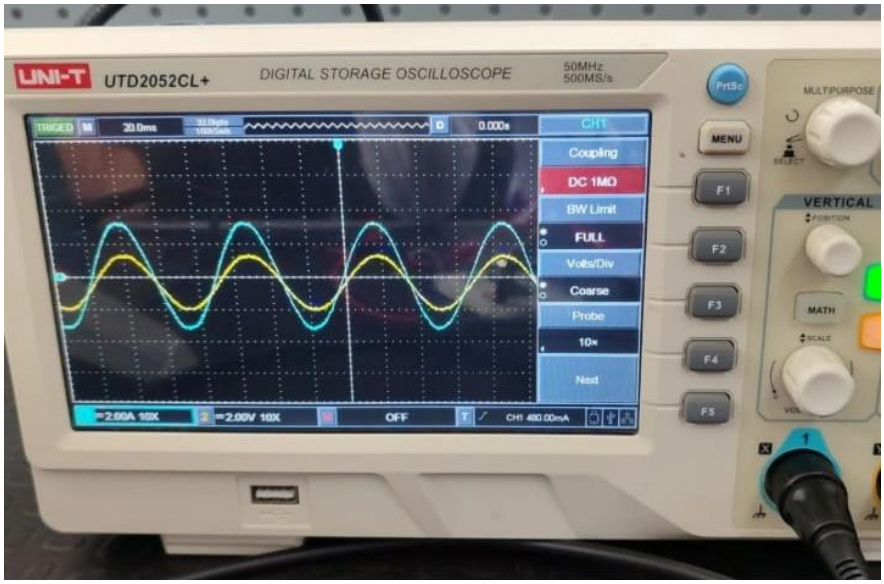
En esta sección, se demuestra la funcionalidad de cada parte electrónica del asistente. Se realizan pruebas y se registran los resultados.

6.1.1. Validación del circuito electrónico

Se analiza la funcionalidad del circuito para la medición del sensor de pulso. Para eso, se conecta el circuito del sensor a un osciloscopio para verificar que el circuito limpia la señal y la analiza correctamente. La señal de entrada se muestra de color azul y la salida se muestra de color amarillo. Se conecta la entrada del filtro, una señal ruidosa y sin procesar, a una entrada del osciloscopio y se conecta la salida del filtro LM358 a otra entrada del osciloscopio para hacer la comparación entre ambas.

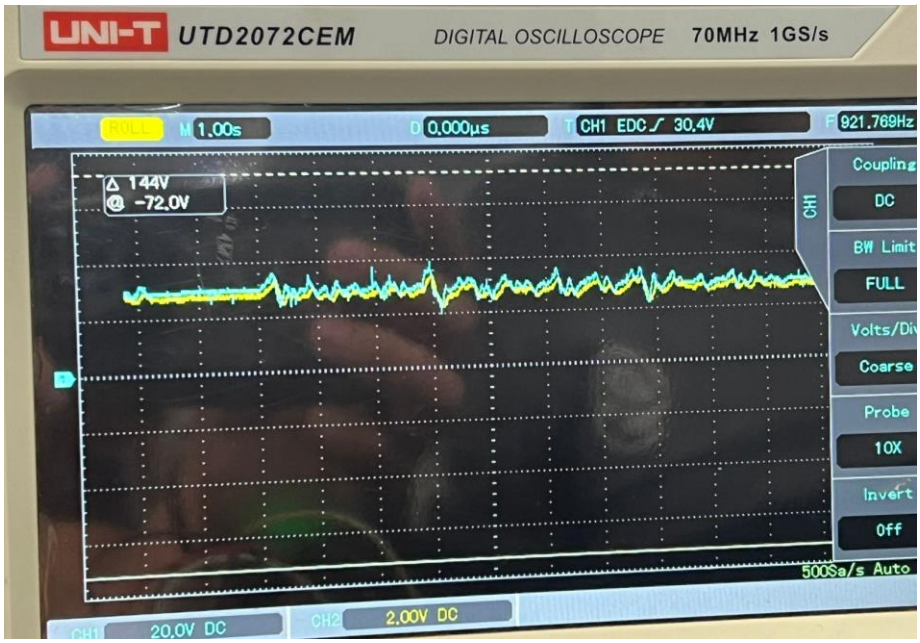
Primero, se conecta un generador de señales para comprobar que el circuito cumple su función de atenuar la señal y dejar pasar ciertas frecuencias. En la Ilustración 6.1. se muestra el análisis del circuito.

Ilustración 6.1. Análisis del circuito con generador de señales



En la imagen se comprueba que el filtro si atenúa la señal por lo que se comprueba que el filtro funciona correctamente. Ahora, se realiza el mismo análisis, solo que se cambia el generador de señales por el sensor de pulso. En la Ilustración 6.2. se muestra el análisis del circuito con el sensor conectado.

Ilustración 6.2. Análisis del circuito con el sensor conectado



NOTA: La señal se ve pequeña debido al ajuste del osciloscopio.

Al igual que con el generador, se ve una señal más ruidosa y con varios picos. Luego, al pasar por filtro, se obtiene una señal más adecuada y ajustada al sistema. De esta manera se confirma la funcionalidad del circuito del sensor de pulso.

6.1.2. Validación del sensor de pulso

Para su validación, se realizan cinco mediciones del sensor de pulso desde el software de la Raspberry Pi y se compara con un oxímetro comercial para tener una referencia. Las cinco mediciones se realizan a 5 personas distintas

para variar en las mediciones. En la Tabla 6.1. se muestran las pruebas realizadas para la verificación de funcionamiento del sensor de pulso.

Tabla 6.1. Pruebas del sensor de pulso

PACIENTE	OXÍMETRO COMERCIAL [BPM]	SENSOR DE PULSO DEL ASISTENTE [BPM]
1	86	88
2	79	77
3	90	93
4	70	69
5	82	80

Con estos datos, se calcula una tasa de efectividad del 97,6% y así, se verifica que el sensor presenta mediciones adecuadas.

6.1.3. Validación del sensor de temperatura

Para la validación del sensor de temperatura se procede de la misma manera que con el sensor de pulso. Se realizan cinco mediciones desde el software de la Raspberry Pi y se compara con un termómetro comercial. En la Tabla 6.2. se muestran las mediciones registradas y la comparación de resultados.

Tabla 6.2. Pruebas del sensor de temperatura

PACIENTE	TERMÓMETRO COMERCIAL	SENSOR DE TEMPERATURA
1	35,9	35,45
2	35,2	34,75
3	36,5	36,64
4	36,3	35,78
5	36,4	36,12

Con estos datos se detecta un error promedio de 0,312°C. Esto indica que el sensor de temperatura que se utiliza en el sistema funciona correctamente, por lo que sus mediciones que registra también pueden tomarse en cuenta para un análisis médico.

6.2. Pruebas de integración y rendimiento del sistema

En esta sección se realizan pruebas del sistema completo para verificar si cumple con los requerimientos que se establecieron anteriormente.

6.2.1. Prueba de rendimiento de software

Para evaluar el rendimiento del software se verifica la conectividad con la inteligencia artificial y el tiempo de respuesta que presenta. Al momento de ejecutar el programa, se debe evaluar que la respuesta sea generada por inteligencia artificial, para eso, se opta por mostrar un mensaje de retroalimentación cada que el sistema llame a la IA para ser usada. En la Ilustración 6.3. se muestra el mensaje que aparece en la terminal del sistema.

Ilustración 6.3. Mensaje de retroalimentación de la inteligencia artificial

```
🔍 Intentando conectar con modelos de Gemini...
   Modelos disponibles a probar: 6

   [1/6] Probando modelo: models/gemini-2.5-flash
   ⏳ Enviando prompt a la IA...
   ✅ ¡RESPUESTA RECIBIDA DEL MODELO models/gemini-2.5-flash!
   📊 Longitud de respuesta: 664 caracteres

📄 VISTA PREVIA DE LA RESPUESTA DE LA IA:
   ----------------------------------------------------------------
   DIAGNÓSTICO: Sospecha de Infección Viral Aguda con Taquicardia
   NIVEL_URGENCIA: ALTA URGENCIA (Nivel Rojo)
   RECOMENDACIÓN: Acudir a urgencias para evaluación médica inmediata y manejo de la fiebre y taquicardia.
   DETALLES: La temperatura corporal de 39.0°C y la taquicardia de 118 bpm son signos de alar...
   ----------------------------------------------------------------

🔬 PROCESANDO RESPUESTA DE LA IA...
   ✓ Diagnóstico extraído: Sospecha de Infección Viral Aguda con Taquicardia
   ✓ Nivel de urgencia: ALTA URGENCIA (Nivel Rojo)
   ✓ Recomendación extraída: Acudir a urgencias para evaluación médica inmediat...

✅ ANÁLISIS COMPLETADO EXITOSAMENTE
   • Diagnóstico: Sospecha de Infección Viral Aguda con Taquicardia
   • Urgencia: ALTA URGENCIA (Nivel Rojo)
   • Timestamp: 2025-11-12 17:12:47
```

Se realizan 10 pruebas diferentes para analizar el tiempo de espera y la conectividad con IA. También, se obtiene un promedio de tiempo de espera. De la misma forma, se corrobora que el sistema cumpla con los requerimientos. En la Tabla 6.3. se muestran los 10 análisis realizados y los tiempos registrados.

Tabla 6.3. Tiempo de espera de respuesta del sistema y conexión con IA

PACIENTE	CONEXIÓN CON IA	TIEMPO DE RESPUESTA [seg]
1	Correcto	15,87
2	Correcto	8,86
3	Correcto	20,39
4	Correcto	16,86
5	Correcto	6,83
6	Correcto	24,58
7	Correcto	41,07
8	Correcto	34,12
9	Correcto	13,80
10	Correcto	7,98
TIEMPO PROMEDIO [seg]		19,04

Con estas pruebas, se obtiene que cada paciente debe esperar un promedio de 19,04 segundos para obtener sus resultados. Además, se corrobora que la integración con IA este concretada.

6.2.2. Prueba de interacción con la interfaz gráfica

Para esta prueba se evalúa la interacción del asistente con cinco personas ajenas al desarrollo del prototipo y sin conocimientos en el área de implementación. En esta prueba se analiza si existe algún tipo de inconveniente u observación al usar el asistente. En la Tabla 6.4. se muestran las pruebas realizadas.

Tabla 6.4. Prueba de interacción del asistente

PACIENTE	RESULTADO ESPERADO	OBSERVACIÓN O INCONVENIENTE
1	Positivo	Ninguno
2	Positivo	Ninguno
3	Positivo	Configuración del teclado muy pequeña
4	Positivo	Tamaño de los botones muy pequeños
5	Positivo	Ninguno

De esta forma se comprueba que el asistente no presenta inconvenientes a la hora de hacer un diagnóstico a los pacientes. Para corregir las observaciones, se aumenta el tamaño del teclado y los botones desde la programación.

6.3. Prueba de precisión del módulo de diagnóstico

El objetivo de esta sección es cuantificar la precisión del análisis hecho con inteligencia artificial del asistente. Se compara el diagnóstico con los de un profesional médico con vasta experiencia en el área de salud. Para esto, se toma en cuenta la opinión de un médico cirujano con 25 años de experiencia en el área de emergencias y en atención primaria en Santa Cruz.

En esta sección, se toman 10 casos con datos reales de pacientes en emergencias. Se realiza un diagnóstico por parte del médico y otro por parte del asistente, para así verificar y evaluar que las respuestas de una inteligencia artificial estén alineadas con las de un profesional de salud.

6.3.1. Elaboración de los casos de prueba

Los casos de prueba se escogen de libros, revistas y páginas web de centros médicos a nivel nacional e internacional. En la Tabla 6.5. se muestra el detalle de cada uno de los casos seleccionados para el análisis.

Tabla 6.5. Casos de prueba

PACIENTE	FUENTE	SINTOMAS	ANTECEDENTES	PULSO	TEMP
1	Casos de la Revista Española de Cardiología y guías de SEMES	Dolor en el pecho atípico, náuseas, sudoración fría, molestias en el pecho	Diabetes tipo 2	105	36,9
2	Guías de Atención Primaria	Dolor de cabeza, congestión nasal, tos seca	Sin antecedentes, alergia a la penicilina	78	37,5

3	Casos clínicos de Sociedad Española de Neumología	Fiebre, tos (con flema), dificultad leve para respirar	Paciente con EPOC (Enfermedad Pulmonar Obstructiva Crónica).	110	38,6
4	Casos clínicos de Revista de Neurología	Dolor de cabeza, náuseas, vómitos, rigidez de nuca	Hipertensión Arterial (HTA) mal controlada.	100	37,2
5	Casos clínicos de Infectología	Fiebre, dolor de cabeza, sensibilidad a la luz, náuseas	Ninguno	115	39,7
6	Guías de Atención Primaria	Dolor de cabeza, estrés, cansancio	Paciente sano, episodios similares en el pasado.	70	36,8
7	Protocolos de Dengue de la OPS/OMS	Fiebre, dolor muscular, dolor detrás de los ojos, náuseas, dolor abdominal leve	Vive en Santa Cruz (zona endémica)	95	38,9
8	Guías de Atención Primaria	Diarrea, náuseas, vómitos	Ninguno	90	37,9
9	Guías de Medicina de Urgencias y Alergología	Dificultad para respirar, hinchazón de labios y lengua, ronchas en la piel, nauseas	Alergia al Ibuprofeno (tomo uno)	130	37,0
10	Casos de Endocrinología y Medicina Interna	Mucha sed, orina frecuente, náuseas, confusión leve	Diabetes tipo 1	115	37,1

6.3.2. Ejecución de diagnóstico

Se ingresan los datos de cada uno de los pacientes de los casos de prueba en el asistente y se anotan los resultados obtenidos. En la Tabla 6.6. se detallan los resultados del asistente.

Tabla 6.6. Diagnóstico del asistente

PACIENTE	URGENCIA	DIAGNÓSTICO	RECOMENDACIONES
1	ALTA	Síndrome coronario agudo	Evaluación médica urgente
2	BAJA	Infección respiratoria leve	Reposo e hidratación
3	ALTA	Exacerbación aguda	Acudir a un médico urgente

4	ALTA	Irritación meníngea	Acudir inmediatamente a emergencias
5	ALTA	Irritación meníngea	Acudir a emergencias
6	BAJA	Cefalea tensional y fatiga	Reposo, hidratación y manejo del estrés
7	MEDIA	Dengue clásico presuntivo	Pruebas diagnósticas y monitoreo
8	MEDIA	Gastroenteritis aguda	Hidratación y buscar valoración medica
9	ALTA	Reacción alérgica severa	Evaluación y manejo de alergia
10	ALTA	Posible cetoacidosis diabética (CAD)	Acudir a emergencias para evaluación y tratamiento

6.3.3. Comparación de diagnóstico

Una vez se cuenta con los resultados del asistente, se los compara con los diagnósticos brindados por el médico. De esta manera, se obtiene un mejor análisis de la efectividad del diagnóstico.

Se realizan dos comparaciones; en la primera se compara el nivel de emergencia que representa tanto para el médico como para el asistente. En la Tabla 6.7. se muestra la comparación.

Tabla 6.7. Comparación del nivel de emergencia de los casos de prueba

PACIENTE	ANÁLISIS EXPERTO	ANÁLISIS ASISTENTE	RESULTADO
1	ALTA	ALTA	ACERTADO
2	BAJA	BAJA	ACERTADO
3	MEDIA	ALTA	ERRADO
4	ALTA	ALTA	ACERTADO
5	ALTA	ALTA	ACERTADO
6	BAJA	BAJA	ACERTADO
7	MEDIA	MEDIA	ACERTADO
8	BAJA	MEDIA	ERRADO
9	ALTA	ALTA	ACERTADO
10	ALTA	ALTA	ACERTADO

En la comparación se detectaron dos falsos positivos, el asistente sobreestimo el riesgo en comparación al experto, lo que resulta en una exactitud y precisión del 80 %.

En la segunda comparación, se analizan los diagnósticos en base al caso. Se registran ambos diagnósticos y se analizan los resultados de cada uno. En la Tabla 6.8. se muestra la comparación.

Tabla 6.8. Comparación del diagnóstico de los casos de prueba

PACIENTE	DIAGNÓSTICO EXPERTO	DIAGNÓSTICO ASISTENTE	RESULTADO
1	Síndrome Coronario Agudo	Síndrome coronario agudo	ACERTADO
2	Rinofaringitis viral o resfrío	Infección respiratoria leve	ACERTADO
3	Exacerbación infecciosa de EPOC o Bronquitis bacteriana.	Exacerbación aguda	ACERTADO
4	Hemorragia subaracnoidea o meningitis	Irritación meníngea	ACERTADO
5	Meningitis bacteriana o viral	Irritación meníngea	ACERTADO
6	Cefalea tensional por estrés	Cefalea tensional y fatiga	ACERTADO
7	Dengue clásico.	Dengue clásico presuntivo	ACERTADO
8	Gastroenteritis aguda viral o alimentaria.	Gastroenteritis aguda	ACERTADO
9	Anafilaxia por AINE	Reacción alérgica severa	ACERTADO
10	Cetoacidosis diabética incipiente.	Posible cetoacidosis diabética (CAD)	ACERTADO

Si bien los términos no son los mismos, el diagnóstico realizado por parte del asistente tiene coherencia con el diagnóstico del médico, por lo que, en esta comparación, se presenta una efectividad del 100 %.

6.4. Evaluación de resultados

En esta sección se analizan los resultados cuantitativos y cualitativos obtenidos de las pruebas de hardware, software y validación diagnóstica detalladas en las secciones anteriores.

6.4.1. Análisis del hardware

Se comprobó la funcionalidad de los componentes electrónicos.

- **Acondicionamiento del circuito electrónico:** La validación del circuito del filtro demostró su efectividad. Como se observa en la Ilustración 6.2. del osciloscopio, la señal del sensor presentó un ruido predominante y picos altos, mientras que la señal de salida del filtro Sallen-Key fue más estable y atenuó las frecuencias por encima de la frecuencia de corte de 4 Hz.

- **Precisión de los sensores:** Las pruebas de validación arrojaron métricas de error aceptables. El sensor de temperatura MLX90614 registro un error promedio de ±0,312°C en comparación al termómetro comercial de referencia. El sensor de pulso, luego del filtro, mostró una precisión del 97.6 % comparado con el oxímetro comercial. Estas variaciones en ambos sensores se deben a la calidad de los componentes utilizados ya que los dispositivos comerciales son elaborados de forma profesional.

6.4.2. Análisis del software y rendimiento

La integración del software y la interfaz gráfica fue exitosa.

- **Tiempo de respuesta:** El sistema cumplió con el requerimiento del diseño. Las pruebas de rendimiento registraron un tiempo de respuesta promedio de 19,04 segundos.
- **Conectividad con IA:** Cada caso fue exitosamente analizado y diagnosticado por medio de IA. Esto se confirma mediante el aviso de retroalimentación que se activa cada que se llama a la API de Gemini en el código del programa.
- **Usabilidad:** Las pruebas de interacción con 5 usuarios indicaron que el flujo de la interfaz es intuitivo. Los usuarios presentaron inconvenientes menores al momento de ingresar sus síntomas, de todos modos, dichos inconvenientes fueron corregidos en el código.

6.4.3. Análisis de la precisión diagnóstico

Esta fue la prueba más crítica del prototipo, ya que cumple la función de validación médica. Como se mostró anteriormente, se validó el asistente

usando 10 casos clínicos reales comparados contra el diagnóstico de un experto médico.

Primero se realizó una comparación del nivel de riesgo que representa para cada uno donde se obtuvo un acierto del 80 %. En la segunda comparación, se midió el nivel de exactitud del diagnóstico médico, donde se obtuvo un 100 % de efectividad. En esta, si bien los términos médicos varían, la respuesta del prototipo está alineada con la opinión médica.

Se promedian ambos porcentajes y se obtiene que el sistema alcanzó una tasa de operación general del 90 %, con un total 18 sobre 20 en aciertos clínicos.

- **Aciertos:** El asistente demostró alta coherencia médica, identificando correctamente 18 de 20 circunstancias médicas, tomando en cuenta los 10 análisis de riesgo y los 10 diagnósticos.
- **Análisis de fallas:** Se detectaron 2 fallas. Ambas fallas se traducen como falsos positivos. El asistente sobreestimó las circunstancias y dio un nivel de riesgo mayor al que corresponde. Este error se atribuye totalmente a la inteligencia artificial y si bien no fue exacto, el error no tiene un impacto de gravedad que pueda haber afectado al paciente.

A pesar de las fallas detectadas, una precisión del 90 % demuestra que el modelo de IA esta significativamente alineado con la perspectiva médica. Sin embargo, se reitera que el asistente sirve como apoyo y que no reemplaza en ningún caso al diagnóstico y evaluación de un profesional de salud.

CAPÍTULO 7

CONCLUSIONES Y RECOMENDACIONES

7. CONCLUSIONES Y RECOMENDACIONES

7.1. Conclusiones

El presente proyecto logró su objetivo de diseño e implementación de un asistente para diagnóstico médico usando inteligencia artificial, enfocado en optimizar el triaje en las salas de emergencia de los hospitales públicos en Bolivia. Tras las fases de diseño, implementación y validación, se ha llegado a las siguientes conclusiones que responden a los objetivos específicos planteados:

- Se caracterizó la situación actual de las salas de emergencia en Bolivia. Se identificó como problemática central la sobrecarga del personal médico y los largos tiempos de espera, los cuales se deben a una alta demanda y a la falta de sistemas eficientes. Esta investigación confirmó la necesidad de una herramienta tecnológica de apoyo.
- Se caracterizaron y seleccionaron los modelos de inteligencia artificial. Se determinó que Gemini, una IA con modelo de procesamiento de lenguaje natural y de aprendizaje profundo, era la más adecuada para el proyecto, ofreciendo una alta fiabilidad y una accesibilidad sencilla en comparación a otras opciones.
- Se diseñó el asistente para diagnóstico médico. Se logró una integración exitosa de hardware y software mediante una Raspberry Pi 4 modelo B. Se destaca el diseño de un circuito de acondicionamiento de señal mediante un filtro activo Sallen-Key de segundo orden y un conversor ADC MCP3008 para procesar la señal analógica del sensor de pulso COMIMARK. El sistema se complementó con un sensor de temperatura MLX90614 y una interfaz gráfica intuitiva mostrada a través de una pantalla HMTECH de 10.1 pulgadas.

- Se implementó el prototipo funcional con base a los requerimientos establecidos. Se desarrolló un circuito electrónico para el acondicionamiento del sensor de pulso. Se imprimió en 3D la carcasa para juntar todos los componentes utilizados en el sistema. Se desarrollo en Python todo el código para el funcionamiento del programa y con Tkinter se desarrolló la interfaz. Luego, se utilizó la API de Gemini como inteligencia artificial.

Por último, las pruebas de rendimiento arrojaron un tiempo de respuesta promedio de 19.036 segundos hasta la entrega del diagnóstico. La validación de precisión diagnóstica, al comparar con una opinión médica, demostró una tasa de acierto del 90 %.

7.2. Recomendaciones

A partir de los resultados obtenidos durante el desarrollo de este proyecto, se proponen las siguientes recomendaciones para mejorar el sistema y orientar futuros trabajos en el área.

- Mejorar el modelo de inteligencia artificial, de esta forma se obtiene un diagnóstico mucho más exacto y se reduce el riesgo de posibles errores o malos diagnósticos. Se recomienda entrenar un modelo de IA o elegir otra opción de inteligencia artificial que esté completamente enfocada en el área médica.
- Implementar un modelo de IA local, así se elimina la dependencia a una conexión de red y reduciría el tiempo de respuesta. Además, se amplía el alcance que puede tener el prototipo, ya que no todos los hospitales cuentan con una conexión estable.
- Realizar pruebas clínicas en entornos reales para evaluar el rendimiento del asistente en entornos reales, aunque, bajo estricta

supervisión del personal médico. Asimismo, se obtiene una retroalimentación del personal médico de emergencias.

- Implementar una base de datos más profesional con base al hospital donde se aplique. Una vez se pruebe en entornos reales, se puede cambiar el sistema para que trabaje con una base de datos específica donde tenga más acceso a la información de los pacientes que ingresan.
- Aplicar más sensores que permitan medir otros valores que también se deben tomar en cuenta en un diagnóstico de emergencia. La frecuencia cardiaca, la presión o la saturación de oxígeno, son algunos datos que también podrían tomarse en cuenta para un mejor análisis.

El presente trabajo sienta las bases para desarrollos posteriores y más avanzados, abriendo así, la posibilidad de crear nuevas tecnologías que beneficien tanto a los pacientes como al personal médico.

CAPÍTULO 8

IMPACTO SOCIAL Y ECONÓMICO

8. IMPACTO SOCIAL Y ECONÓMICO

El desarrollo de un asistente para diagnóstico médico controlado mediante inteligencia artificial representa un avance significativo en la inclusión de tecnologías para la optimización del sistema de salud público en Bolivia. El presente proyecto representa un aporte significativo al campo de la salud digital y al desarrollo de tecnologías de asistencia funcional, particularmente en las salas de emergencia que enfrentan una alta demanda y escasez de recursos.

Más allá de la implementación de un sistema de apoyo automatizado, este trabajo abre posibilidades para la exploración y validación de nuevas formas de interacción paciente-sistema que optimicen el flujo de atención y la asignación de recursos médicos. Además, al haber logrado resultados funcionales con herramientas de libre acceso, como la plataforma Raspberry Pi, y hardware de bajo costo diseñado a medida, el proyecto contribuye a ampliar el acceso a tecnologías de diagnóstico inteligente, tradicionalmente limitadas a equipos importados de alto costo, y demuestra que la innovación en tecnología médica también puede desarrollarse desde contextos académicos locales.

Este trabajo no solo aporta a la solución de un problema técnico y operativo, sino que contribuye al desarrollo de una sociedad más inclusiva, equitativa y consciente del valor de la tecnología como medio para mejorar la calidad de vida y la eficiencia de los servicios de salud.

CAPÍTULO 9

ANEXOS

9. ANEXOS

9.1. Código completo del asistente en Python

Por motivos de extensión y practicidad, el código completo del asistente, compuesto por aproximadamente 1300 líneas de código en Python, no se ha adjuntado en forma impresa en este documento.

El código, que incluye la implementación de la interfaz gráfica con Tkinter, la lógica de adquisición de datos de los sensores de temperatura MLX90614 y el convertidor MCP3008, la conexión a la base de datos usando SQLite y las llamadas a la API de la inteligencia artificial Gemini descritas en el Capítulo 5, se encuentra disponible para su consulta y revisión en el siguiente repositorio digital:

https://github.com/andres122333/Proyecto-de-grado---Asistente-medico-con-IA---Andres-Ignacio-Justiniano-Ortega

9.2. Hoja de datos del sensor MLX90614

Melexis
Microelectronic Integrated Systems

MLX90614 family
Single and Dual Zone
Infra Red Thermometer in TO-39

Ordering Information

Part No.	Temperature Code	Package Code	- Option Code
MLX90614	E (-40°C to 85°C) K (-40°C to 125°C)	SF (TO-39)	- X X X (1) (2) (3)

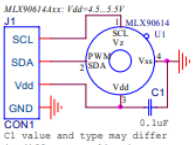
(1) Supply Voltage/ Acccuracy
A - 5V
B - 3V
C - Reserved
D - 3V medical accuracy

(2) Number of thermopiles:
A – single zone
B – dual zone

(3) Package options:
A – Standard package
B – Reserved
C – 35° FOV

Example:
MLX90614ESF-BAA

1 Functional diagram



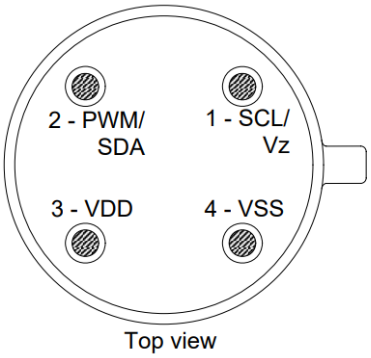
MLX90614 connection to SMBus

Figure 1 Typical application schematics

2 General Description

The MLX90614 is an Infra Red thermometer for non contact temperature measurements. Both the IR sensitive thermopile detector chip and the signal conditioning ASSP are integrated in the same TO-39 can.
Thanks to its low noise amplifier, 17-bit ADC and powerful DSP unit, a high accuracy and resolution of the thermometer is achieved.
The thermometer comes factory calibrated with a digital PWM and SMBus (System Management Bus) output.
As a standard, the 10-bit PWM is configured to continuously transmit the measured temperature in range of -20 to 120 ˚C, with an output resolution of 0.14 ˚C and the POR default is SMBus.

6 Pin definitions and descriptions



Pin Name	Function
VSS	Ground. The metal can is also connected to this pin.
SCL / Vz	Serial clock input for 2 wire communications protocol. 5.7V zener is available at this pin for connection of external bipolar transistor to MLX90614A to supply the device from external 8 -16V source.
PWM / SDA	Digital input / output. In normal mode the measured object temperature is available at this pin Pulse Width Modulated. In SMBus compatible mode automatically configured as open drain NMOS.
VDD	External supply voltage.

Table 2: Pin description MLX90614

9.3. Hoja de datos del convertidor ADC MCP3008

MCP300X 10-Bit Analog-to-Digital Converters

Product Information

The Microchip MCP300X family of 10-bit analog-to-digital converters (ADCs) combines high performance and low power consumption in a small package – making it ideal for embedded control applications.

Consisting of the MCP3001, MCP3002, MCP3004 and MCP3008, the MCP300X family features a successive approximation register (SAR) architecture and an industry-standard SPI™ serial interface. Devices are available with 1, 2, 4 or 8 input channels and in PDIP, SOIC and TSSOP packages.

The MCP300X family offers existing Microchip customers added flexibility when incorporating analog inputs into their designs. The industry standard SPI interface allows 10-bit ADC capability to be added to any PICmicro®

Additional Information:

- Microchip's web site: www.microchip.com
- Microchip's *Technical Library CD-ROM*, Order No. DS00161
- More than 112 Application Notes available:
  - *Embedded Control Handbook*, Order No. DS00092
  - *Embedded Control Handbook, Volume 2, Math Library*, Order No. DS00167
- Microchip's *Overview, Quality Systems and Customer Interface System*, Order No. DS00169
- Third party software and hardware support:
  - Emulators
  - Programmers
  - Gang Programmers
  - Software Tools
  - Development Boards and Accessories
  - Design Consultants
  - *Third Party Guide*, Order No. DS00104

MCP300X High-Performance 10-Bit Analog-to-Digital Converters

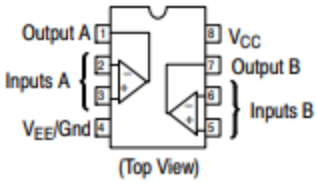
Product	Resolution (Bits)	No. of Channels	Sampling Rate (ksps)	INL (±LSB)	DNL (±LSB)	Supply Voltage	Temperature Range	Standby Current @ 5V (typical, µA)	Operating Current @ 5V (typical, µA)	Packages
MCP3001	10	1	200	1	1	2.7 - 5.5	-40° to +85°C	0.005	400	8P, 8SO, 8TSSOP
MCP3002	10	2	200	1	1	2.7 - 5.5	-40° to +85°C	0.005	525	8P, 8SO, 8TSSOP
MCP3004	10	4	200	1	1	2.7 - 5.5	-40° to +85°C	0.005	425	14P, 14SO, 14TSSOP
MCP3008	10	8	200	1	1	2.7 - 5.5	-40° to +85°C	0.005	425	16P, 16SO

9.4. Hoja de datos del amplificador operacional LM358

Features

- Short Circuit Protected Outputs
- True Differential Input Stage
- Single Supply Operation: 3.0 V to 32 V
- Low Input Bias Currents
- Internally Compensated
- Common Mode Range Extends to Negative Supply
- Single and Split Supply Operation
- ESD Clamps on the Inputs Increase Ruggedness of the Device without Affecting Operation
- NCV Prefix for Automotive and Other Applications Requiring Site and Control Changes
- These Devices are Pb−Free, Halogen Free/BFR Free and are RoHS Compliant

PIN CONNECTIONS



MAXIMUM RATINGS (T_A = +25°C, unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltages Single Supply Split Supplies	 V_{CC} V_{CC}, V_{EE}	 32 ±16	Vdc
Input Differential Voltage Range (Note 1)	V_{IDR}	±32	Vdc
Input Common Mode Voltage Range (Note 2)	V_{ICR}	−0.3 to 32	Vdc
Output Short Circuit Duration	t_{SC}	Continuous	
Junction Temperature	T_J	150	°C
Thermal Resistance, Junction−to−Air (Note 3) Case 846A Case 751 Case 626	$R_{\theta JA}$	238 212 161	°C/W
Storage Temperature Range	T_{stg}	−65 to +150	°C
ESD Protection at any Pin Human Body Model Machine Model	V_{esd}	 2000 200	V
Operating Ambient Temperature Range LM258 LM358, LM358A LM2904/LM2904A LM2904V, NCV2904 (Note 4) NCV2904V (Note 4)	T_A	 −25 to +85 0 to +70 −40 to +105 −40 to +125 −40 to +150	°C

CAPÍTULO 10

REFERENCIAS BIBLIOGRÁFICAS

10. REFERENCIAS BIBLIOGRÁFICAS

[1] M. d. Salud, Norma Nacional de Caracterizacion de Hospitales de Segundo Nivel, La Paz: Documentos Tecnicos Normativos, 2014.

[2] C. Healthcare, «Cuidado de urgencia vs. sala de emergencias,» Cigna Healthcare, 2025. [En línea]. Available: https://www.cigna.com/es-us/knowledge-center/urgent-care-vs-emergency-room.

[3] M. Zambrano, «Salas de emergencia,» Scribd, 11 Agosto 2011. [En línea]. Available: https://es.scribd.com/doc/61394287/Salas-de-emergencia-2.

[4] Unitel, «Por falta de personal cierran sala de urgencias en el Hospital Japonés de Santa Cruz,» Unitel, Agosto 2023. [En línea]. Available: https://noticias.unitel.bo/salud/por-falta-de-personal-cierran-sala-de-urgencias-en-el-hospital-japones-de-santa-cruz-BC8756829.

[5] C. M. Flores, «Crisis en el Hospital Oncológico: Atención saturada y condiciones críticas,» Red Uno, 22 Agosto 2024. [En línea]. Available: https://www.reduno.com.bo/noticias/crisis-en-el-hospital-oncologico-atencion-saturada-y-condiciones-criticas-2024822125447.

[6] Defensoria, «ASUSS confirma deficiencias en los servicios de emergencias de hospitales obreros de la CNS identificadas por la Defensoría del Pueblo,» Defensoria del Pueblo , 15 Julio 2022. [En línea]. Available: https://www.defensoria.gob.bo/noticias/asuss-confirma-deficiencias-en-los-servicios-de-emergencias-de-hospitales-obreros-de-la-cns-identificadas-por-la-defensoria-del-pueblo.

[7] F. Milenio, «Salud en crisis: pacientes esperan hasta un mes por consultas y cirugías,» El Deber, 2023.

[8] Medienta, «Motivos de atención en salas de emergencia en hospitales publicos de La Paz Bolivia,» Scielo, 22 Abril 2016. [En línea]. Available: http://www.scielo.org.bo/scielo.php?script=sci_arttext&pid=S1652-67762016000200002.

[9] J. C. Torrejon, «Los hospitales de Santa Cruz, Bolivia, están saturados por los casos de dengue,» EFE, 14 Febrero 2023. [En línea]. Available:

https://efe.com/salud/2023-02-14/los-hospitales-de-santa-cruz-bolivia-estan-saturados-por-los-casos-de-dengue/.

[10] O. L. Salazar, «Santa Cruz con más de 15.051 afectados por conjuntivitis y otras enfermedades a causa de los incendios,» El Dia, 5 Octubre 2024. [En línea]. Available: https://www.eldia.com.bo/2024-10-05/santa-cruz/santa-cruz-con-mas-de-15051-afectados-por-conjuntivitis-y-otras-enfermedades-a-causa-de-los-incendios.html.

[11] INE, «Censo de poblacion y vivienda.,» Instituto de Nacional y Vivienda, 2012. [En línea]. Available: https://www.ine.gob.bo/index.php/censos-salud/.

[12] M. d. Salud, «Ministerio de Salud Promueve Uso de la Tecnología Con TELESALUD,» Ministerio de Salud, 22 Abril 2015. [En línea]. Available: https://www.minsalud.gob.bo/544-ministerio-de-salud-promueve-uso-de-la-tecnologia-con-telesalud.

[13] A. H. Alvez, Tecnología médica y políticas., Pando: Universidad Amazonica de Pando, 2024.

[14] M. d. Salud, «Ministerio de Salud Promueve Uso de la Tecnologia con TELESALUD,» MinSalud, 22 Abril 2015. [En línea]. Available: https://www.minsalud.gob.bo/544-ministerio-de-salud-promueve-uso-de-la-tecnologia-con-telesalud.

[15] Oporto, Recomendaciones para la modernización de las salas de emergencia en Bolivia, Boletin de Innovacion en Salud Publica, 2022.

[16] PAHO/OPS, «Perfil del Pais - Bolivia,» Organizacion Panamericana de la Salud, 15 Septiembre 2024. [En línea]. Available: https://hia.paho.org/es/perfiles-de-pais/bolivia.

[17] Prodware, «Inteligencia Artificial,» Prodware, 2025. [En línea]. Available: https://www.prodwaregroup.com/es-es/innovacion/inteligencia-artificial/.

[18] Telefonica, «Que es y como funciona la inteligencia artificial,» Telefonica, 25 Noviembre 2022. [En línea]. Available: https://www.telefonica.com/es/sala-comunicacion/blog/que-es-y-como-funciona-la-inteligencia-artificial/.

[19] Cloud, « ¿Qué es la inteligencia artificial o IA?,» Google Cloud, 2025. [En línea]. Available: https://cloud.google.com/learn/what-is-artificial-intelligence?hl=es-419.

[20] Tableau, «Tipos de inteligencia artificial,» Tableau, 2025. [En línea]. Available: https://www.tableau.com/es-mx/data-insights/ai/tipos-de-inteligencia-artificial#:~:text=Esta%20clasificaci%C3%B3n%20considera%20tre.

[21] IBM, «¿Que es un modelo de IA?,» IBM, 13 Septiembre 2023. [En línea]. Available: https://www.ibm.com/es-es/think/topics/ai-model.

[22] QuestionPro, «Modelos de inteligencia artificial: Que es, tipos y el papel de los datos sinteticos,» QuestionPro, 30 Octubre 2023. [En línea]. Available: https://www.questionpro.com/blog/es/modelos-de-inteligencia-artificial/.

[23] C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[24] HPE, «Modelos de IA,» Hewlett Packard Enterprise, 2021. [En línea]. Available: https://www.hpe.com/lamerica/es/what-is/ai-models.html.

[25] UDIT, «¿Qué es el procesamiento del lenguaje natural (NLP)?,» Universidad de Diseño, Innovacion y Tecnologia, 8 Enero 2024. [En línea]. Available: https://udit.es/actualidad/que-es-el-procesamiento-del-lenguaje-natural/.

[26] U. Europea, «Inteligencia artificial en medicina: revolucion de la salud digital,» Universidad Europea, 2 Julio 2024. [En línea]. Available: https://universidadeuropea.com/blog/inteligencia-artificial-medicina/.

[27] Sanofi, «Usos de la inteligencia artificial en medicina y sus beneficios en la salud de los pacientes,» Campus Sanofi, 4 Mayo 2024. [En línea]. Available: https://pro.campus.sanofi/es/actualidad/articulos/inteligencia-artificial-salud.

[28] GeeksforGeeks, «Applications of Deep Learning In Healthcare,» GeeksforGeeks, 6 Junio 2024. [En línea]. Available: https://www.geeksforgeeks.org/deep-learning/applications-of-deep-learning-in-healthcare/?utm_source.

[29] A. Arif, «How Natural Language Processing Is Helping Doctors Make Better Diagnoses,» John Snow Labs, 10 Mayo 2023. [En línea]. Available:

https://www.johnsnowlabs.com/how-natural-language-processing-is-helping-doctors-make-better-diagnoses/?utm_source.

[30] I. Roche, «Inteligencia artificial en salud: Retos éticos y legales.,» Instituto Roche, 2024. [En línea]. Available: https://www.institutoroche.es/static/archivos/Informes_anticipando_.

[31] S. Salas, «Consideraciones éticas respecto de la Inteligencia Artificial,» Bioeticar, 2022. [En línea]. Available: https://medicina.udd.cl/centro-bioetica/files/2022/05/Salas-SP-2022.-Inteligencia-Artificial.pdf.

[32] A. G. Lopez, La integración de la inteligencia artificial, Colombia: Universitas Medica, 2023.

[33] M. Electronics, «Raspberry Pi 4 Modelo B,» MCI Electronics, 26 Septimbre 2025. [En línea]. Available: https://raspberrypi.cl/producto/raspberry-pi-4b-4gb-ram/.

[34] U. Electronics, «Raspberry PI 4 Modelo B (4GB | 8GB) RAM,» UNIT Electronics, 2025. [En línea]. Available: https://uelectronics.com/producto/raspberry-pi-4-modelo-b-version-4gb-8gb-ram/?srsltid=AfmBOor6IiyoMM6GjmLPJ9lBbaO8myFwC4sta3pGnu3RZphgCSnl7rMp.

[35] L. CATEDU, «Electrónica de Raspberry Pi 4 modelo B,» Libreria CATEDU, 2025. [En línea].

[36] R. P. Documentation, «Raspberry Pi computer hardware,» Raspberry Pi Documentation, 2025. [En línea]. Available: https://www.raspberrypi.com/documentation/computers/raspberry-pi.html#gpio.

[37] Pacifiko, «Pantalla HMTECH,» Pacifiko, 2025. [En línea]. Available: https://www.pacifiko.com/compras-en-linea/hmtech-10-1-inch-raspberry-pi-screen-hdmi-monitor-1024x600-lcd-display-16-9-ips-screen-for-raspberry-pi-4-3-2-zero-b-b-win11-10-8-7-free-driver&pid=Yjk1YzIxZT?srsltid=AfmBOopPP9wfcJBQ8ea-wtRc2NTpoV-AmQ_xEqbaZuq4vL.

[38] T. Shop, «Sensor de Temperatura Infrarrojo GY-906 MLX90614,» Tostatronic Shop, 2025. [En línea]. Available: https://www.tostatronic.com/product/sensor-de-temperatura-infrarrojo-gy-906-mlx90614/?srsltid=AfmBOorQnpTlxGT5yZhEjwAD16eOcQVA_wUk6z5OYBrHdvVSCSRwgqdL.

[39] Amazon, «Comimark Módulo sensor de pulso de ritmo cardíaco,» Amazon, 2025. [En línea]. Available: https://www.amazon.com/Comimark-Sensor-Module-Arduino-Raspberry/dp/B07V6VV8CM.

[40] MathWorks, «Introduccion a los filtros paso bajo,» MathWorks, 2025. [En línea]. Available: https://la.mathworks.com/discovery/low-pass-filter.html.

[41] L. A. Electronics, «Filtro Paso Bajo,» Learning About Electronics, 2025. [En línea]. Available: https://www.learningaboutelectronics.com/Articulos/Filtro-paso-bajo.php.

[42] U. Electronics, «MCP3008-I/P Convertidor Analógico a Digital DIP-16,» UNIT Electronics, 2025. [En línea]. Available: https://uelectronics.com/producto/mcp3008-i-p-convertidor-analogico-a-digital-dip16/?srsltid=AfmBOopaK_UblJ2GJOdJuUBlcXzTmWC0OTXGBTAQz7GnMLFIELHse3I_.

[43] HETECH, «10.1 inch 1024x600 capactive touch HDMI display,» HETECH, 2025. [En línea]. Available: https://www.sz-htc.com/products_details/24.html.

[44] L. M. Engineers, «Interface MLX90614 Non-contact Infrared Temperature Sensor with Arduino,» Last Minute Engineers, 2025. [En línea]. Available: https://lastminuteengineers.com/mlx90614-ir-temperature-sensor-arduino-tutorial/.

[45] M. Plus, «Signos vitales,» Medline Plus, 1 Enero 2025. [En línea]. Available: https://medlineplus.gov/spanish/ency/article/002341.htm.

[46] i. Electronics, «Amplificador LM358,» i2C Electronics, 2025. [En línea]. Available: http://i2celectronica.com/amplificadores/613-amplificador-lm358.html.

[47] M. d. Salud, Norma Nacional de Caracterizacion de Hospitales de Segundo Nivel, La Paz: Documentos Tecnicos Normativos, 2014.

[48] MICROCHIP, «Convertidor MCP3008,» Amazon, 2025. [En línea]. Available: https://www.amazon.com/-/es/Microchip-mcp3008-i-convertidor-anal%C3%B3gico-digital/dp/B011NA05PA.

[49] T. L. Floyd, Dispositivos Electronicos, Pearson Education, 2008.

[50] CIEESC, «Honorarios y aranceles para ingenieros,» Colegio de Ingenieros Electricistas y Electrónicos de Santa Cruz, 2017. [En línea]. Available: https://cieesc.com/.

CURRÍCULUM VITAE

ANDRES IGNACIO JUSTINIANO ORTEGA

(+591) 75371839

andresjustinianoortega@gmail.com

FORMACIÓN ACADÉMICA

2018	Certificación de Ingles. TOEFL ITP.
	Centro Boliviano Americano.
2019	Certificación de Ingles. ELASH II.
	College Board.
2007 - 2020	Bachiller en Humanidades.
	Colegio Mayor Santo Tomás de Aquino.
2021 – presente	Licenciatura en Ingeniería Electrónica.
	Universidad Privada de Santa Cruz de la Sierra.

PRACTICAS LABORALES

Febrero 2024 – presente

Ayudante de Laboratorio de Electrónica.

Universidad Privada de Santa Cruz de la Sierra.

HABILIDADES TECNICAS

Manejo de Instrumentos de medición electrónicos.

Diseño y montaje de circuitos electrónicos.

Manejo de software Proteus

Manejo básico de paquetes de Office

Manejo de software y hardware ARDUINO

Conocimientos de programación en Python.

Conocimientos de modelado 3D en software Fusion 360

PREMIOS Y RECONOCIMIENTOS

2017	Primer Lugar en las Olimpiadas de Matemáticas de la gestión 2017. Colegio Mayor Santo Tomas de Aquino.
2014 - 2019	Diploma al Mérito por excelencia académica durante las gestiones escolares. Colegio Mayor Santo Tomas de Aquino.
2021	Segundo Lugar en la competencia Tecno UPSA. Universidad Privada de Santa Cruz de la Sierra.

ACTIVIDADES ADICIONALES

2020	Certificado de Participación en el curso Pre Universitario. Universidad Privada de Santa Cruz de la Sierra.
2020	Certificado de Participación del curso "Programación y Matemáticas. Universidad Privada de Santa Cruz de la Sierra.
2022	Miembro del STAFF del evento Lacnic 38. LACNIC.
2023	Certificado de Participación en la Promo UPSA 2023. Universidad Privada de Santa Cruz de la Sierra.
2023	Participación como miembro del STAFF de la 1ra Olimpiada de Robótica Educativa. Universidad Privada de Santa Cruz de la Sierra.
2024	Participación como miembro del STAFF de la 2da Olimpiada de Robótica Educativa. Universidad Privada de Santa Cruz de la Sierra.
2025	Participación como miembro del STAFF de la 3ra Olimpiada de Robótica Educativa. Universidad Privada de Santa Cruz de la Sierra.