



UNIVERSIDAD TECNOLÓGICA ISRAEL
ESCUELA DE POSGRADOS “ESPOG”

MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN
Resolución: RPC-SO-09-No.265-2021

PROYECTO DE TITULACIÓN EN OPCIÓN AL GRADO DE MAGÍSTER

Título del proyecto:
Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal.
Línea de Investigación:
Ciencias de la ingeniería aplicadas a la producción, sociedad y desarrollo sustentable
Campo amplio de conocimiento:
Ingeniería, industria y construcción
Autor:
Leonardo Alexander Flores Gutiérrez
Tutor/a:
Ing. Wilmer Albarracín

Quito – Ecuador

2025

APROBACIÓN DEL TUTOR 1



Yo, **Wilmer Fabian Albarracin Guarochico** con C.I: **17133413352** en mi calidad de Tutor del proyecto de Investigación titulado: **Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal. Previo a la obtención del título de Magister en Electrónica y automatización.**

Elaborado por: **Ing. Leonardo Alexander Flores Gutiérrez** de C.I. **0503502809**, estudiante de la Maestría: Electrónica y Automatización de la **UNIVERSIDAD TECNOLÓGICA ISRAEL(UISRAEL)**, como parte de los requerimientos sustanciales con fines de obtener el Título de Magister, me permito declarar que luego de haber orientado, analizado y revisado el trabajo de titulación, lo apuro en todas su partes.

Quito D.M., 15 de septiembre de 2025



Firma

APROBACIÓN DEL TUTOR 2



Yo, **_René Ernesto Cortijo Leyva_** con C.I.: **_1719010108_** en mi calidad de Tutor del proyecto de investigación titulado: **Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal.**

Elaborado por: **_Leonardo Alexander Flores Gutiérrez_** de C.I.: **_0503502809_**, estudiante de la Maestría: **_Electrónica y Automatización_**, de la **UNIVERSIDAD TECNOLÓGICA ISRAEL (UISRAEL)**, como parte de los requisitos sustanciales con fines de obtener el Título de Magister, me permito declarar que luego de haber orientado, analizado y revisado el trabajo de titulación, lo apruebo en todas sus partes.

Quito D.M., 30 de septiembre del 2025



Firma

DECLARACIÓN DE AUTORIZACIÓN POR PARTE DEL ESTUDIANTE



Yo, **Flores Gutiérrez Leonardo Alexander** con C.I: **0503502809** autor/a del proyecto de titulación denominado: **Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal**. Previo a la obtención del título de Magister en Electrónica y automatización.

1. Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar el respectivo trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
2. Manifiesto mi voluntad de ceder a la Universidad Tecnológica Israel los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor@ del trabajo de titulación, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital como parte del acervo bibliográfico de la Universidad Tecnológica Israel.
3. Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de prosperidad intelectual vigentes.

Quito D.M., 30 de septiembre de 2025



Leonardo Alexander
Flores Gutierrez
Time Stamping
Security Code

Firma

Tabla de contenidos

APROBACIÓN DEL TUTOR	2
DECLARACIÓN DE AUTORIZACIÓN POR PARTE DEL ESTUDIANTE	3
Tabla de contenidos	4
Índice de tablas	7
Índice de figuras	8
Índice de anexos	10
INFORMACIÓN GENERAL	11
Contextualización del tema	11
Objetivo General	13
Objetivos específicos	13
Vinculación con la sociedad y beneficiarios directos	14
CAPÍTULO I: DESCRIPCIÓN DEL PROYECTO	15
1.1 Contextualización general del estado del arte	15
1.2 Proceso investigativo metodológico	17
CAPÍTULO II: PROPUESTA	19
2.1 Fundamentos teóricos aplicados	19
2.1.1 Monitoreo de señales biomédicas	19
2.1.2 Sensores biomédicos	19
2.1.3 Sensor max30102	20
2.1.4 Frecuencia cardíaca del ser humano	21
2.1.5 Saturación de oxígeno en sangre (SpO ₂)	22
2.1.6 Enfermedades relacionadas al BPM y SpO ₂	23
2.1.7 Sensor de temperatura	24
2.1.8 Temperatura corporal	24
2.1.8.1 Enfermedades relacionadas con la temperatura	25
2.1.9 Anomalías del ser humano	26
2.1.10 Tarjeta Argon particle	26

2.1.11 Internet de las Cosas (IoT)	27
2.1.12 Red neuronal Keras	28
2.1.13 Almacenamiento en la nube (Firebase)	29
2.1.14 Ubidots	30
2.1.15 LCD	31
2.1.16 Batería de litio	32
2.2 Descripción de la propuesta	32
a. Estructura general	33
b. Explicación del aporte	34
c. Estrategias y/o técnicas	36
2.2.1 Desarrollo del prototipo	36
2.2.2 Plataforma de configuración de tarjeta Argon	38
2.2.2.1 Consola de particle web IDE	38
2.2.2.2 Barra de herramientas web IDE	39
2.2.2.3 Integraciones IoT de plataforma particle	40
2.2.2 Programación de tarjeta Argon	41
2.2.3 Comunicación con Firebase y Ubidots	43
2.2.3.1 Configuración de Webhook para Firebase y ubidots	43
2.2.3.2 Configuración de la base de datos	44
2.2.3.3 Configuración de Ubidots	46
2.2.4 Red neuronal Keras	46
2.5.5 Desarrollo de aplicación para la PC	52
2.3 Validación de la propuesta	54
2.4 Matriz de articulación de la propuesta	56
2.5 Análisis de resultados. Presentación y discusión	58
2.5.1 Análisis de resultados del sistema	58
2.5.3 Rendimiento de la red neuronal	68
CONCLUSIONES	72

RECOMENDACIONES	74
REFERENCIAS BIBLIOGRÁFICAS	75
ANEXOS	80

Índice de tablas

Tabla 1	Valores normales de frecuencia cardíaca en reposo por grupo etario	22
Tabla 2	Valores normales y umbrales críticos de SpO2:	23
Tabla 3	Valores normales y umbrales críticos de SpO2	25
Tabla 4	Perfil de validadores	54
Tabla 5	Escala de evaluación de Ph.D David Rivas	55
Tabla 6	Escala de evaluación Msc. Gissela Guanoluisa	55
Tabla 7	Escala de evaluación Ing. Walter Villarroel	55
Tabla 8	Matriz de articulación	56
Tabla 9	Muestra de las mediciones de variables para validar el sistema.	62
Tabla 11	Muestra del cálculo de eficiencia y error de cada paciente	66
Tabla 11	Condiciones médicas de enfermedades relacionadas a las variables medidas	70

Índice de figuras

Figura 1	<i>Arquitectura general de un sensor utilizado en sistemas IoT</i>	20
Figura 2	<i>Sensor Max 30102</i>	21
Figura 2	<i>Sensor de temperatura</i>	24
Figura 4	<i>Tarjeta Argon.</i>	27
Figura 5	<i>Clasificación de la librería Tensor Flow y Keras dentro las APIs.</i>	28
Figura 6	<i>Modelo de red neuronal Keras de una neurona</i>	29
Figura 7	<i>Descripción general de la arquitectura de Firebase</i>	30
Figura 8	<i>Pantalla LCD.</i>	31
Figura 9	<i>Batería de litio.</i>	32
Figura 4	<i>Diagrama del sistema de monitoreo y análisis.</i>	33
Figura 11	<i>Diagrama esquemático del prototipo</i>	37
Figura 12	<i>Circuito impreso del prototipo</i>	37
Figura 13	<i>Equipo desarrollado del proyecto</i>	38
Figura 14	<i>Plataforma particle web IDE</i>	39
Figura 15	<i>Barra de herramientas de particle IDE</i>	40
Figura 16	<i>Integraciones IoT de particle</i>	40
Figura 17	<i>Diagrama de flujo de adquisición de señales de sensores</i>	41
Figura 18	<i>Librerías cargadas al programa</i>	42
Figura 19	<i>Código de lectura de sensores</i>	42
Figura 20	<i>Configuración para comunicación con Firebase</i>	43
Figura 21	<i>Configuración para comunicación con Firebase y ubidots</i>	44
Figura 22	<i>Diagrama de comunicación con la nube</i>	45
Figura 23	<i>Configuración de base de datos en Firebase</i>	45
Figura 24	<i>Dashboard de las variables en ubidots</i>	46
Figura 25	<i>Dashboard de las variables en ubidots</i>	47
Figura 26	<i>Dashboard de las variables en ubidots</i>	47
Figura 27	<i>Modelo de aprendizaje profundo secuencial de la red Keras</i>	50
Figura 28	<i>Programación de red neuronal para el aprendizaje profundo</i>	51
Figura 29	<i>Entropía de la red neuronal, iteraciones</i>	51
Figura 28	<i>Archivos del programa desarrollado para la PC</i>	52
Figura 31	<i>Visualización de datos del dispositivo Particle ARGON en el aplicativo.</i>	53

Figura 32	<i>Muestras de sistema local y remoto.</i>	58
Figura 33	<i>Curva de variable de temperatura medidas y predicha</i>	58
Figura 34	<i>Curva de variable de SpO2 medidas y predicha</i>	59
Figura 35	<i>Exactitud y pérdida de red neuronal</i>	60
Figura 36	<i>Gráficas de la red neuronal BPM de un paciente gráfica de Ubidots</i>	60
Figura 37	<i>Probabilidad de la predicción paciente según la red</i>	61
Figura 38	<i>Predicción de la red para 1 paciente</i>	61
Figura 39	<i>Matriz confusión de los datos de la red</i>	62
Figura 40	<i>Matriz confusión de los datos de la red</i>	62
Figura 41	<i>Error promedio de BPM por paciente</i>	64
Figura 42	<i>Error promedio SpO2</i>	65
Figura 43	<i>Error promedio Temperatura</i>	65
Figura 44	<i>Error promedio por variable</i>	67
Figura 45	<i>Eficiencia promedio por variable</i>	68
Figura 46	<i>Desempeño de la Red Neuronal.</i>	69
Figura 47	<i>Consumo de energético del prototipo</i>	71

Índice de anexos

Anexo 1 Distribución de pines de la tarjeta de control particle Argon	80
Anexo 2 Diagrama del prototipo pcb en el programa PCB Wizard	80
Anexo 3 Implementación física del prototipo	81
Anexo 4 Programación de tarjeta Argon particle, de la red neuronal Keras en Google colab, y programación del aplicativo.	81
Anexo 5 Pacientes participantes del proyecto	90
Anexo 6 Artículos para proyectos de medicina para poder ser probados en hospitales	90

INFORMACIÓN GENERAL

El seguimiento de los signos vitales constituye una herramienta esencial tanto en la prevención como en el tratamiento de diversas enfermedades, al posibilitar la detección temprana de cambios en la salud de los pacientes. En la historia estos sistemas se han restringido en recintos hospitalarios, lo que complica la monitorización continua de pacientes con enfermedades crónicas o en condiciones de riesgo según lo mencionado por (Shaik, y otros, 2023). Ahora se tiene como prioridad el desarrollo de las tecnologías que faciliten la calidad de vida como es la introducción del Internet de las Cosas (IoT), lo que da lugar al desarrollo dispositivos capaces de recopilar y transmitir datos en tiempo real, aumentando así la accesibilidad y la efectividad de realizar monitoreo a distancia.

Un reto importante en el monitoreo de los signos vitales es el análisis de grandes paquetes de entrada de información para identificar de forma automática patrones anómalos. En este sentido, utilizar la Inteligencia Artificial y técnicas de aprendizaje automático se convierte en una estrategia clave para optimizar la detección de irregularidades, disminuyendo los falsos positivos y aumentando la precisión en las decisiones de un diagnóstico o tratamiento. Las plataformas como Firebase ofrecen una infraestructura segura y escalable para almacenar datos o información, que posteriormente se puede analizar, facilitando el acceso a estos datos a los profesionales de salud y permitiendo la implementación de sistemas de alerta automáticos.

El estudio desarrollado en este trabajo tiene como finalidad desarrollar un sistema de visualización de signos vitales, que utilice la combinación de IoT e IA, utilizando una tarjeta Argon de Particle y Firebase para almacenar datos en tiempo real y con la ayuda de una RN identificar patrones anómalos. Se espera que este sistema potencie la eficiencia del seguimiento remoto de pacientes y ofrezca una solución tecnológica innovadora para el ámbito médico. La integración de IoT, IA y almacenamiento en la nube representa una estrategia prometedora para la modernización del monitoreo de salud, facilitando intervenciones clínicas más rápidas y precisas (SEMERGEN), 2021).

Contextualización del tema

Se conoce que actualmente, el monitoreo continuo de signos vitales puede ser de tiempo real es una forma muy acertada para la crucial atención oportuna de personas en condiciones vulnerables. Al utilizar el monitoreo especialmente en pacientes que han sufrido accidentes de tránsito, laborales u otros, que presentan un alto riesgo de deterioro repentino de su salud, así como en personas que padecen de hipertensión, hipotensión, hipoxia o fiebre por largos períodos, donde cualquier alteración de los signos puede dar como resultado una emergencia que no se pueda

detectar a tiempo lo que agravaría la situación del paciente. En zonas rurales o entornos de difícil acceso a una casa de salud donde ocurre la emergencia, la posibilidad de monitorear remotamente el estado de una persona puede significar la diferencia entre la vida y la muerte al no tener una atención médica oportuna.

Los equipos tradicionales usados para el monitoreo de un paciente suelen depender de instrumentos especializados instalados en hospitales, lo cual restringe que sean llevados a lugares fuera del hospital, y se sabe que para operarlos se requiere de personal capacitado, lo cual representa gasto para el sistema de salud. Por lo que en la actualidad se toma como una opción el desarrollo de proyectos de bajo coste que permitan monitorear remotamente a un paciente. Sin embargo, uno de los desafíos es que se pueda tener un diagnóstico acertado o la detección temprana de patrones anómalos que pueden afectar a la salud, evitando los falsos positivos que puedan alarmar innecesariamente a los usuarios o médicos.

En el contexto sobre la integración de IoT con tecnologías que tenga IA permite el desarrollo de sistemas autónomos capaces de analizar y almacenar grandes cantidades de datos o información ya sea biomédica o de otra índole en tiempo real. Por lo que el presente proyecto propone un sistema basado en una tarjeta Argon Particle para lograr la adquisición continua de datos como es la frecuencia cardíaca, saturación del oxígeno en la sangre y temperatura del cuerpo, los cuales son transmitidos a través de la nube por medio de IoT. A su vez, se analiza los mismos datos por una red neuronal entrenada basada en la red Keras para detectar automáticamente alguna anomalía en el paciente lo que ayuda a actuar de manera oportuna.

Por lo que la implementación de un prototipo de tales características resulta especialmente útil cuando se requiera el monitoreo de pacientes hipertensos o hipertensos, seguimiento de adultos mayores con enfermedades relacionadas o para personas que hayan sufrido un accidente. Para asegurar su funcionalidad, el sistema también considera aspectos críticos como la latencia de transmisión de información, el procesamiento y la seguridad de los datos de cada paciente. De esta forma, se plantea una solución escalable, segura y efectiva, alineada con las necesidades tanto del paciente y de los galenos, acorde a lo que menciona la Organización Mundial de la Salud (OMS) y además lo que está establecido en el Ministerio de Salud Pública del Ecuador cuando se desarrolla este tipo de proyectos.

Problema de investigación

En la actualidad los equipos de monitoreo de variables de cualquier índole, aún persisten desafíos significativos como en la fiabilidad, precisión y tiempo de muestreo lo que suele ser una

desventaja frente a los sistemas convencionales, en cambio estos suelen requerir intervención humana constante y no siempre están diseñados para procesar datos instantáneamente. Y son muy pocos o casi nada los equipos que integran inteligencia artificial lo que limita su capacidad para identificar patrones anómalos de manera inmediata y precisa. Esta situación puede derivar en diagnósticos tardíos o imprecisos, afectando la salud del paciente y aumentando el riesgo de alguna complicación que pueda provocar una condición crítica. Por lo tanto, surge la necesidad de investigar y construir un sistema que permita la recolección y análisis de señales biomédicas mediante tecnologías IoT e IA, optimizando la detección de anomalías sin comprometer la fiabilidad de la información.

El presente estudio aborda esta problemática para plantear el desarrollo de un prototipo basado en una tarjeta Argon de Particle para la comunicación con Firebase y el uso de modelos de IA para la identificación de anomalías en la salud de las personas. Se investiga cómo la implementación de algoritmos de aprendizaje, puede mejorar la precisión de los diagnósticos y reducir la incidencia en agravar la salud de una persona, garantizando una respuesta más eficiente ante posibles emergencias médicas. Asimismo, se explorarán estrategias para optimizar la transmisión, almacenamiento y procesamiento de los datos, asegurando la escalabilidad y seguridad del sistema. A partir de este enfoque, se espera contribuir al desarrollo de proyectos que sean innovadores para la atención médica remota y la prevención de complicaciones en pacientes vulnerables.

Objetivo

Objetivo General

Desarrollar un sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal.

Objetivos específicos

- Definir los requerimientos técnicos y funcionales del sistema de monitoreo de signos vitales, incluyendo los parámetros fisiológicos a medir, las tecnologías IoT a utilizar y los criterios clínicos de detección de anomalías.
- Diseñar un sistema IoT funcional, basado en la tarjeta Argon Particle y sensores biomédicos adecuados, que permita la captura y transmisión continua de datos de frecuencia cardíaca, temperatura corporal y saturación de oxígeno.

- Desarrollar un algoritmo para entrenar la red neuronal Keras de TensorFlow, para la detección de anomalías en la salud de pacientes relacionados con los signos vitales mediante aprendizaje automático.
- Integrar la red neuronal entrenada con el entorno de visualización en Visual Studio Code, utilizando p5.js para representar gráficamente los resultados en tiempo real y mostrar alertas automatizadas ante desviaciones clínicas.
- Validar el funcionamiento del prototipo mediante pruebas de simulación y casos clínicos controlados, evaluando su precisión en la detección de anomalías, latencia de transmisión y fiabilidad operativa en entornos de monitoreo remoto

Vinculación con la sociedad y beneficiarios directos

El presente proyecto establece un vínculo directo con la sociedad mediante la implementación de un sistema de monitoreo remoto de signos vitales en el Centro de Salud de San Buenaventura, con el propósito de optimizar el seguimiento clínico de pacientes en riesgo y mejorar la capacidad de respuesta del personal médico ante eventos críticos. Este sistema, basado en tecnologías IoT, inteligencia artificial y visualización en tiempo real, representa una solución tecnológica avanzada, pero de bajo costo, adaptable a las necesidades reales del entorno local.

San Silvestre es una parroquia del barrio San Buenaventura de la ciudad de Latacunga que cuenta con una población estimada de entre 2000 y 3.000 habitantes ya que es una zona rural, quienes acceden a servicios médicos de primer nivel a través del centro de salud de la zona considerado como tipo C. Se estima que al menos 150 a 200 pacientes mensuales presentan condiciones clínicas que requieren monitoreo continuo o periódico de alguna señal del cuerpo, tales como hipertensión arterial, enfermedades respiratorias, entre otras. Por lo que estas personas podrían ser los beneficiarios directos del sistema, ya que podrán ser monitoreadas de forma más precisa, automatizada y oportuna, incluso fuera del subcentro de salud, pero solo se tomó como prueba clínica controlada ya que el prototipo debe ser evaluado por MSP para que pueda ser utilizado de manera formal. Entre los beneficiarios indirectos, se incluyen familiares de los pacientes monitoreados, personal administrativo del centro de salud y otros usuarios eventuales del sistema.

CAPÍTULO I: DESCRIPCIÓN DEL PROYECTO

1.1 Contextualización general del estado del arte

En los últimos años, el monitoreo de signos vitales en tiempo real ha experimentado una evolución significativa gracias al desarrollo e integración de las tecnologías emergentes como es el IoT, el almacenamiento en la nube y la IA. Estas tecnologías han permitido diseñar sistemas capaces de recopilar, transmitir y analizar datos fisiológicos con alta precisión y de forma continua, generando así herramientas que fortalecen el diagnóstico temprano y la respuesta médica inmediata. Entre los parámetros más monitoreados se encuentran la frecuencia cardíaca, la saturación del oxígeno en sangre de una persona y la temperatura del cuerpo, cuya alteración puede reflejar condiciones médicas críticas. Diversos estudios desde 2019 han abordado la eficacia de estos sistemas, destacando su aplicación en entornos clínicos, domiciliarios y rurales.

Las tecnologías IoT aplicadas a la medicina son varias entre las cuales se destacan los sensores biométricos integrados en microcontroladores y tarjetas controladoras las que permiten la recolección de datos biomédicos como lo menciona Andrade (2024) en su trabajo, se evaluaron plataformas basadas en microcontroladores con conexión a Wifi y Bluetooth, principalmente para tarjetas ESP32 y Arduino con módulos de comunicación inalámbrica, a las cuales se acoplaron sensores ópticos para medir frecuencia cardíaca y oxigenación de la sangre, así como sensores de temperatura infrarrojos y de contacto. Estos dispositivos fueron evaluados por el autor en un entorno clínico controlado con un grupo reducido de voluntarios, donde los datos recopilados se enviaron a Firebase y ThingsBoard, lo que permitió analizar la información en tiempo real. (Andrade, 2024).

Por su parte, Chen et al. (2024), en un estudio publicado por la revista Nature, desarrolló una arquitectura inalámbrica que permitió monitorear simultáneamente temperatura, pulso y presión arterial de una persona, demostrando que el uso de dispositivos con tecnología IoT facilita la vigilancia no invasiva y constante en cualquier lugar, reduciendo la carga de trabajo hospitalaria lo que mejoraría la oferta de equipos. El estudio que realiza tiene un enfoque orientado a poblaciones vulnerables y de tercera edad. Debido a que el dispositivo tiene conexión wi fi hace posible que se pueda utilizarlo en cualquier lugar con acceso a la red, además que el tráfico de datos es instantáneo, también se menciona que los resultados obtenidos tienen una correlación superior del 95% de los valores registrados con otro equipo lo que evidenció la fiabilidad de los datos obtenidos.

Desde el punto de vista de la precisión de los datos Efendi (2025), en su trabajo propone un sistema de monitoreo remoto específicamente para adultos mayores basado en la plataforma Firebase y algoritmos XGBoost, La validación del sistema se realizó en un grupo piloto conformado por adultos mayores con condiciones de diagnóstico reservado, donde los resultados demostraron una precisión del 97,3 % en la clasificación y detección de eventos anómalos, además de una reducción significativa de falsos positivos en comparación con otros algoritmos de aprendizaje automático. Estas métricas evidencian que la combinación de las herramientas es robusta para la vigilancia continua de este grupo poblacional.

El análisis de datos con inteligencia artificial es abordado por Rosca et al. (2025) el cual propone un modelo híbrido compuesto por redes neuronales de memoria a corto y largo plazo con modelos convolucionales, este proyecto analiza los registros de signos vitales reales de personas hospitalizadas, se logró detectar anomalías con una precisión superior al 97 % y redujo en un 20 % las readmisiones hospitalarias por eventos prevenibles por la red neuronal desarrollada en este trabajo, por lo que el uso de modelos de machine learning mejora sustancialmente la capacidad predictiva de los sistemas de monitoreo remoto, al identificar patrones no evidentes para el ojo humano, por lo que mejora la atención médica preventiva (Rosca, 2025).

La visualización y presentación de resultados de un proyecto de adquisición de señales como lo menciona, Ghadi (2025) destaca el papel de herramientas como p5.js en el diseño de interfaces interactivas y visuales que permiten al personal médico y a los pacientes interpretar los datos con mayor facilidad. Este enfoque ha sido replicado en varios proyectos de salud pública y ciudades inteligentes, donde la interfaz juega un papel fundamental en la comprensión de alertas generadas por modelos automatizados. (Ghadi, 2025)

Por otra parte, aunque no existen estudios que mencione directamente el uso de la tarjeta Argon Particle, la literatura académica respalda la elección de microcontroladores con conectividad Wi-Fi, bajo consumo energético y capacidad de transmisión en tiempo real como componentes clave en soluciones de monitoreo biomédico. Bapayya Naidu Kommula (2019) argumenta que la eficiencia de estos dispositivos depende de la calidad de los sensores utilizados, la estabilidad de la red y la estructura de la base de datos que respalda el sistema.

En conjunto, la revisión de estos antecedentes evidencia que monitoreo de señales biomédicas mediante IoT y el análisis con machine learning, ha alcanzado un alto impacto en la medicina, por lo que se tiene que seguir en la innovación en entornos de atención preventiva al paciente dentro o fuera de la casa de salud. El proyecto se inscribe en esta tendencia, al combinar la

tarjeta Argon Particle, Firebase Realtime Database para la adquisición de señales y luego analizarlos en modelos neuronales entrenados en Google Colab y visualización en p5.js, con el propósito de ofrecer una solución integral para realizar el monitoreo remoto de pacientes. Con ello, se contribuye a cerrar la brecha tecnológica en salud, mejorar la detección temprana de anomalías clínicas y mejorar o verificar la toma de decisiones médicas basadas en las muestras.

1.2 Proceso investigativo metodológico

El presente proyecto estudia la necesidad de desarrollar algún sistema de monitoreo y predicción de signos mediante la integración de tecnologías IoT e inteligencia artificial. El objetivo principal es combinar sensores biomédicos, almacenamiento en la nube y modelos de aprendizaje que sea automático para optimizar la detección automática de anomalías de los parámetros fisiológicos críticos. La metodología se estructuró en diversas etapas que abarcan desde la revisión teórica hasta la validación experimental del sistema, empleando enfoques metodológicos específicos para cada fase.

La investigación previa al desarrollo del proyecto, se la hace con el fin de establecer las bases teóricas del proyecto y de cada una de sus etapas. Se analizaron trabajos previos relacionados con el monitoreo de señales biomédicas mediante dispositivos IoT y análisis de los mismos con alguna red neuronal, además de almacenar en la nube u otras plataformas como Firebase, amazon, etc. Esta indagación bibliográfica incluyó artículos científicos, tesis académicas y documentación técnica relacionada con el tema, lo que permitió definir los requerimientos funcionales y tecnológicos del sistema, así como justificar su viabilidad del proyecto.

La implementación de la red neuronal en este proyecto se hace para el análisis de datos biomédicos adquiridos por la tarjeta, previo a esto se la entrenó con una base de datos de Kaggle, que incluye registros de frecuencia cardíaca, saturación de oxígeno en la sangre y la temperatura corporal. Para entrenar la red se debe tener en cuenta el modelo matemático adecuado para este proyecto además los datos deben ser filtrados, normalizados para que la red tenga información clara y precisa. El análisis de los datos permitió garantizar la consistencia y fiabilidad del entrenamiento orientado a cumplir el objetivo del proyecto.

En la etapa de desarrollo e implementación del proyecto, se procedió con la integración de sensores biomédicos a la tarjeta Argon Particle, la cual se encarga de capturar los datos inmediatamente y transmitirlos a Firebase a una base RD para su almacenamiento en la nube en un archivo json. Esta etapa se desarrolló bajo un enfoque de investigación experimental, ya que se fue depurando el código hasta tenerlo con el mínimo de error con el propósito de optimizar la precisión

y estabilidad de la transmisión y recolección de la información. Paralelamente se aplica una solución funcional orientada a resolver una necesidad concreta del sector salud como es el monitoreo remoto. En el entrenamiento de la red neuronal Keras, se utilizó Colab junto con las bibliotecas de Tensor Flow, dado que se evaluaron múltiples arquitecturas de red y se ajustaron diversos parámetros para maximizar la precisión y eficacia del modelo en la detección de anomalías. A su vez, se aplicó pruebas en el modelo fue entrenado para validar los distintos escenarios y verificar su desempeño sobre los datos fisiológicos analizados.

La validación del sistema se realizó mediante pruebas con voluntarios en condiciones controladas, los datos recopilados en tiempo real se compararon con las predicciones del modelo entrenado, evaluando su efectividad a través de métricas y tablas que indican los riesgos en la alteración de las variables adquiridas. En esta fase metodológica se enmarca dentro de una investigación experimental ya que no se cuenta con los permisos suficientes para realizarlos en casas de salud, pero sí con condiciones de control. Adicionalmente, se mantuvo una lógica experimental al ajustar parámetros del modelo según los resultados obtenidos en cada iteración.

Los resultados obtenidos a lo largo de esta investigación ofrecen una solución eficiente y realista para el monitoreo continuo de señales biomédicas por lo que esto ayuda a la salud tanto en entornos clínicos como domiciliarios. Para finalizar se documenta detalladamente el funcionamiento del prototipo, sus beneficios potenciales y su utilidad en el contexto de la medicina digital. Asimismo, mantiene un enfoque sobre cómo se puede mejorar los diagnósticos y prever futuras enfermedades o emergencias, al tener como propósito la solución de un problema real mediante el uso de tecnologías emergentes el proyecto se encuentra en los estándares de investigación.

CAPÍTULO II: PROPUESTA

2.1 Fundamentos teóricos aplicados

2.1.1 Monitoreo de señales biomédicas

Los signos vitales representan parámetros esenciales que permiten evaluar el estado fisiológico de órganos clave como el cerebro, el corazón y los pulmones. Entre estos se incluyen la temperatura corporal, la frecuencia cardíaca, la respiración y la presión arterial. La medición adecuada de estos factores resulta crucial para identificar alteraciones en el organismo, facilitando un diagnóstico ágil y preciso sobre la salud del individuo. Cambios en estos valores pueden ser señales de condiciones médicas como infecciones, problemas cardiovasculares o deshidratación, lo que requiere atención oportuna. (Villegas González, Villegas Arenas, & Villegas González, 2012)

El análisis correcto de los signos vitales es fundamental dentro del ámbito clínico, ya que ayuda a tomar decisiones en el diagnóstico y en el tratamiento del paciente. Por ejemplo, un aumento en la temperatura puede sugerir un caso de infección o inflamación, mientras que variaciones en la presión arterial podrían reflejar alteraciones en el estado del paciente como el estrés, u otras afecciones. Del mismo modo, la saturación del oxígeno en la sangre es clave para detectar el estado circulatorio y pulmonar del paciente. Por este motivo, la evaluación continua de estos signos se considera una herramienta indispensable para garantizar una atención médica eficaz y de calidad en pacientes que hayan sufrido un accidente o tengan una enfermedad. (Rubio, 2011)

En los últimos años, la tecnología de monitorización de signos vitales ha experimentado avances significativos. La integración de herramientas con IA y la conectividad inalámbrica ha permitido el desarrollo de dispositivos portátiles y sistemas de telemetría médica que facilitan el seguimiento continuo y remoto. Por ejemplo, los anillos inteligentes de Oura pueden monitorear parámetros como es la frecuencia cardíaca y la temperatura de la piel, alertando a los usuarios sobre alguna posible anomalía antes de que se presenten síntomas evidentes. (Merino, 2024) Además, la incorporación de sensores avanzados en dispositivos cotidianos, como relojes inteligentes, permite la medición no invasiva de niveles de glucosa en la sangre, lo que representa un avance significativo para personas con diabetes. (HuffPost, 2025)

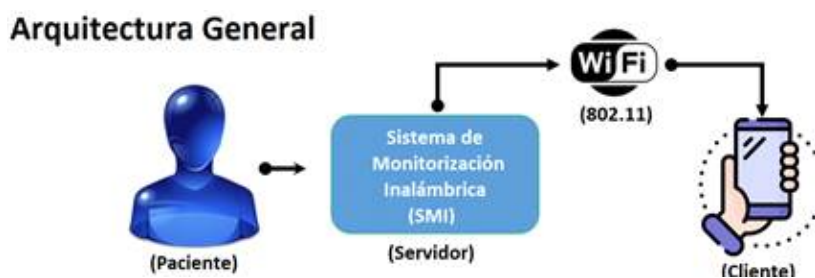
2.1.2 Sensores biomédicos

El desarrollo de sensores biomédicos integrados en sistemas de Internet de las Cosas (IoT) ha permitido avances significativos en la monitorización de la salud en la figura 1 se observa cómo estos

dispositivos recopilan información en tiempo real sobre diferentes parámetros fisiológicos y la transmiten a plataformas digitales para su análisis que es su arquitectura general. Gracias a la conectividad inalámbrica, los datos pueden ser procesados de manera eficiente, lo que facilita diagnósticos más precisos y permite ajustar los tratamientos de forma personalizada. La evolución de estas tecnologías ha dado lugar a sensores más pequeños, con menor consumo energético y con capacidad de adhesión a la piel o de implantación en el organismo, mejorando la comodidad del paciente y optimizando la precisión en la recolección de datos (Baibhab Chatterjee, 2023).

Figura 1

Arquitectura general de un sensor utilizado en sistemas IoT



Nota: Tomada de Smith Colas 2022.

A pesar de sus múltiples beneficios, el uso de sensores biomédicos con IoT enfrenta desafíos importantes. La seguridad de los datos es un aspecto crítico para la casa de salud y el paciente, ya que la información médica es altamente sensible. Además, los dispositivos de distintos fabricantes compiten con mejorar sus prestaciones, así como el ahorro del consumo de energía para garantizar un monitoreo continuo sin interrupciones y menor consumo energético. Al utilizar la IA en el análisis de datos han mejorado la capacidad predictiva de estos sistemas, permitiendo detectar anomalías antes de que se presenten síntomas evidentes. No obstante, es fundamental continuar desarrollando estrategias que aseguren su implementación segura (Baibhab Chatterjee, 2023).

2.1.3 Sensor max30102

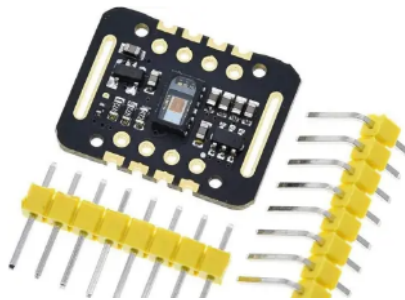
Los sensores se encargan de captar los datos del paciente, dichas señales eléctricas son transformadas a información por la tarjeta que se utiliza para recopilar las señales en tiempo real de todas las variaciones del sensor. Este dispositivo se utiliza fundamentalmente en aplicaciones de monitoreo, ya que permiten medir la saturación del oxígeno en la sangre y las pulsaciones por minuto las cuales son fundamentales para conocer el estado de salud o riesgo de enfermedad que pueda tener una persona en tiempo real. Su funcionamiento se basa en:

- La fotopletismografía permite la lectura de la actividad sanguínea en el dedo para poder tomar las señales eléctricas.
- Su fuente de alimentación es 3.3 Vdc
- Es capaz de medir Bpm y SpO2 para lo cual se necesita una librería.
- Tiene comunicación SCL y Serial.

la conversión de una magnitud física en una señal eléctrica, la cual puede ser procesada por la tarjeta electrónica Argón de Particle que funciona como controlador y realiza la adquisición de las magnitudes de este sensor, como se sabe se debe utilizar una librería que permita la lectura las cuales pueden ser las mismas de Arduino.

Figura 2

Sensor Max 30102



Nota: Tomada de Last Minute Engineers (2025).

El sensor MAX 30102 en la figura 2, mide el pulso y la saturación del oxígeno en la sangre a través de la fotopletismografía, por lo que se necesita una librería cargada en la tarjeta que vaya adquirir la señal de este dispositivo para tener alta precisión en los datos, pero también depende del fabricante como se sabe en el mercado existen sensores mucho más precisos, pero más costosos.

2.1.4 Frecuencia cardíaca del ser humano

Al hablar sobre la frecuencia cardíaca de un individuo se hace referencia al número que el corazón late por un minuto, comúnmente conocida como pulsaciones por minuto. Este parámetro fisiológico es fundamental en la evaluación del estado funcional del sistema cardiovascular, ya que refleja la actividad que realiza el corazón al bombear sangre y oxigenar los tejidos de todo cuerpo de una persona. En condiciones normales, una frecuencia cardíaca en estado de reposo oscila entre 60 hasta los 100 BPM en adultos sanos, aunque puede variar dependiendo de factores como la edad, el sexo, el nivel de condición física, el estado emocional y la presencia de patologías subyacentes como se observa en la tabla 1.

Tabla 1*Valores normales de frecuencia cardíaca en reposo por grupo etario*

Grupo de edad	Rango de BPM en reposo
Recién nacidos 0 a 1 mes	70 hasta los 190
Lactantes (1 a 11 meses)	80 hasta los 160
Niños pequeños (1 a 2 años)	80 hasta los 130
Preescolares (3 a 4 años)	80 hasta los 120
Escolares (5 a 6 años)	75 hasta los 115
Niños mayores (7 a 9 años)	70 hasta los 110
Adolescentes y adultos	60 hasta los 100

Es fundamental tener en cuenta que estos valores pueden variar según el individuo y su condición física. Por lo tanto, cualquier desviación significativa de estos rangos debe ser evaluada por un profesional de la salud. (Schulman, 2021).

Diversos factores pueden influir en la frecuencia cardíaca. El ejercicio físico incrementa temporalmente las pulsaciones debido a la mayor demanda de oxígeno por los músculos. El estrés y las emociones intensas también pueden elevar la frecuencia cardíaca como respuesta del cuerpo a situaciones de tensión, (Doctor Julio Maset, 2020). Mantener una frecuencia cardíaca dentro de los rangos normales es crucial para la salud cardiovascular, frecuencias elevadas o disminuidas pueden indicar problemas cardíacos subyacentes. La detección temprana de estas anomalías permite intervenciones oportunas, reduciendo el riesgo de complicaciones. Por ello, es recomendable monitorear regularmente la frecuencia cardíaca y consultar a un doctor ante cualquier irregularidad. (Valle, 2024).

2.1.5 Saturación de oxígeno en sangre (SpO₂)

El SpO₂ o saturación mide el porcentaje de la hemoglobina arterial saturada con el oxígeno en la sangre, reflejando la eficiencia del transporte del mismo desde los pulmones hacia todo el cuerpo humano. Por lo que mantener niveles adecuados de oxigenación es esencial para el funcionamiento óptimo de órganos y tejidos, ya que el oxígeno es fundamental en los procesos metabólicos del cuerpo, además en la producción de la energía celular de cada persona por que los valores de la saturación se encuentran clasificados en la tabla 2:

Tabla 2

Valores normales y umbrales críticos de SpO₂:

Estado de salud	Saturación de oxígeno (SpO ₂)
Normal	95% hasta los 100%
Hipoxemia leve	90% hasta los 94%
Hipoxemia moderada	80% hasta los 89%
Hipoxemia severa	Menos de 80%

En personas con enfermedades pulmonares crónicas, como la EPOC suelen tener valores de saturación entre 88% y 92% pueden considerarse aceptables según (Jill Seladi-Schulman, 2021).

2.1.6 Enfermedades relacionadas al BPM y SpO₂

Hipoxia. Es el estado en el cual el organismo, o una región específica de este, no recibe una cantidad adecuada de oxígeno para mantener sus funciones. Esta deficiencia puede afectar distintos órganos y tejidos, generando síntomas como la dificultad en la respiración, mareos, confusión en casos más severos, pérdida de conciencia. La hipoxia puede ser provocada por diversas condiciones, como enfermedades pulmonares, altitudes elevadas o problemas circulatorios que impiden una correcta oxigenación de la sangre. Dependiendo de su gravedad y duración, puede derivar en daños irreversibles si no se trata a tiempo (bupasalud, s.f.).

Enfermedad Epoc, Una de las enfermedades es la enfermedad pulmonar obstructiva crónica, el cual es un trastorno respiratorio que se caracteriza por la obstrucción que es persistente del flujo de aire, lo cual hace que se dificulte la respiración y se reduce drásticamente la cantidad de oxígeno presente en la sangre. Esta condición suele desarrollarse debido a la exposición extendida de alguna sustancia irritante, el humo de cigarrillo o algún contaminante ambiental. Los principales síntomas que incluyen son tos crónica, producción excesiva de mucosidad, disnea y fatiga. Si la enfermedad avanza, los episodios de exacerbación pueden aumentar, incrementando el riesgo de insuficiencia respiratoria y el deterioro en la calidad de vida de las personas (Robert A. Wise, 2024).

2.1.7 Sensor de temperatura

El sensor DS18B20 es un termómetro digital de precisión que opera mediante contacto directo con la piel, ideal para aplicaciones biomédicas donde se requiere monitorear la temperatura corporal de forma continua y precisa. Este sensor es un encapsulado de acero inoxidable en la

figura 2, lo cual permite estar en contacto directo con la piel y otras superficies además de ambientes húmedos sin deterioro o alteraciones de la señal, lo que lo hace apto para sistemas de monitoreo clínico o domiciliario.

A nivel técnico, el DS18B20 proporciona:

- Lecturas de temperatura con una precisión de $\pm 0.5\text{ }^{\circ}\text{C}$ en el rango de $-10\text{ }^{\circ}\text{C}$ a $+85\text{ }^{\circ}\text{C}$.
- Rango de operación más amplio de $-55\text{ }^{\circ}\text{C}$ hasta $+125\text{ }^{\circ}\text{C}$.
- Utiliza el protocolo de comunicación digital 1-Wire, lo que permite su conexión con microcontroladores mediante un solo pin de datos (tautvidas, 2015).

Figura 3

Sensor de temperatura



Nota: Tomada de tautvidas (2015).

El DS18B20 se conecta directamente a cualquier tarjeta sin necesidad de un circuito para normalizar la señal. Esta integración facilita la lectura, transmisión y almacenamiento de la temperatura. Debido a su bajo consumo energético es muy adecuado para proyectos portátiles y la facilidad de programación en entornos como Arduino y otras plataformas ya que es compatible con librerías estándar en Python o C, el DS18B20 se posiciona como una herramienta efectiva para la creación de dispositivos biomédicos portátiles, especialmente en entornos con recursos limitados y de bajo coste.

2.1.8 Temperatura corporal

La temperatura corporal es un indicador clave del estado del organismo, reflejando el equilibrio entre la producción y la pérdida de calor del cuerpo. Este equilibrio es mantenido por la termorregulación del organismo el cual es un proceso homeostático que permite al cuerpo adaptarse a variaciones térmicas internas y externas, asegurando un entorno óptimo para las funciones metabólicas. El hipotálamo actúa como el centro regulador principal ya que recibe las señales de los termorreceptores distribuidos en la piel y en órganos internos, por lo que para regular la

temperatura activa respuestas como la sudoración o el escalofrío para mantener la temperatura normal del cuerpo. El rango normal depende de la edad y actividad por lo que el promedio aceptable es de 37°C; sin embargo, estudios recientes indican que este valor puede variar entre 36.1°C y 37.2°C en adultos (DrTango, 2023).

La temperatura de un recién nacido puede oscilar entre 36.1°C y 37.7°C generalmente, mientras que, en adultos mayores puede ser ligeramente más baja debido a la edad. Además, la temperatura corporal suele ser más baja en las horas matutinas y alcanzar su pico en la tarde o noche al momento de dormir puede disminuir (Fidel Ramón-Romero, 2014).

2.1.8.1 Enfermedades relacionadas con la temperatura

La fiebre se puede entender como el aumento de la temperatura del cuerpo por encima de los valores estándares en la medicina, por lo general superando los 38°C cuando se mide oralmente en adultos. (Healthwise, 2024). Este incremento térmico es una respuesta del sistema inmunológico ante infecciones, ya que muchos patógenos tienen una tolerancia limitada a temperaturas elevadas. No obstante, temperaturas superiores a 39.4°C pueden ser motivo de preocupación y requieren atención médica de emergencia. A continuación, en la tabla 3 se resume los rangos normales de temperatura corporal según diferentes grupos etarios:

Tabla 3
Valores normales y umbrales críticos de SpO2

Grupo de edad	Rango de temperatura corporal (°C)
Recién nacidos	36.1 - 37.7
Lactantes	36.1 - 37.7
Niños (2-8 años)	36.1 - 37.7
Adultos	36.1 - 37.2
Adultos mayores	35.8 - 36.9

Estos valores pueden variar según el individuo y el método de medición utilizado ya sea por contacto directo o por infrarrojo. (Tango, 2025)

Es fundamental considerar que la temperatura corporal puede fluctuar debido a diversos factores, y una desviación ocasional no siempre indica una condición patológica. Los cambios

continuos pueden ser síntoma de otras enfermedades por lo que deben ser revisados por un doctor para un diagnóstico adecuado.

2.1.9 Anomalías del ser humano

El monitoreo de los signos es esencial para evaluar la condición de la salud de un individuo ya que pueden indicar parámetros que influyen en la salud del cuerpo humano como, por ejemplo, una elevación en la temperatura del cuerpo puede aumentar la frecuencia cardíaca, mientras la disminución de la presión arterial puede ser que se induzca un incremento en la frecuencia del pulso. Estas relaciones permiten a los galenos poder identificar desequilibrios fisiológicos y responder de manera oportuna (López-Orozco, 2012).

. Algunas anomalías más comunes que presente el ser humano pueden ser las siguientes:

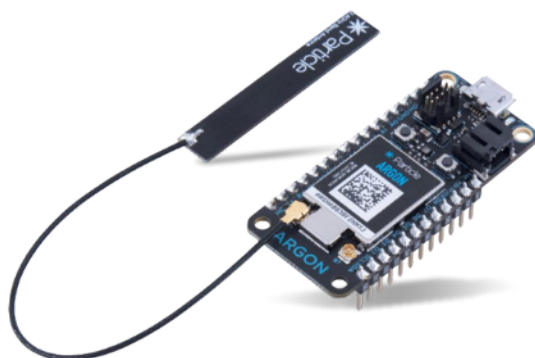
- La temperatura elevada suele ser provocada por una infección.
- El aumento de la presión puede dar por estrés, preocupación o alguna alteración del organismo.
- La temperatura baja puede indicar escalofríos por alguna enfermedad o hipotermia.
- Baja saturación del oxígeno en la sangre indica algún problema cardiovascular o pulmonar (theguardian, 2024).

2.1.10 Tarjeta Argon particle

El Argon de Particle actúa como el núcleo central del sistema automatizado propuesto. Esta tarjeta de desarrollo IoT de alto rendimiento es responsable de procesar la información proveniente de sensores biomédicos, establecer conexiones con servicios en la nube para el monitoreo en tiempo real y comunicarse con el modelo de red neuronal implementado. Su arquitectura está optimizada para aplicaciones que requieren conectividad inalámbrica eficiente, operando mediante Wi-Fi de 2.4 GHz, lo que le permite transmitir datos a una velocidad inferior a 10 bytes por milisegundo. Esta capacidad de transmisión casi instantánea resulta fundamental para sistemas de vigilancia médica remota donde la latencia puede afectar la toma de decisiones clínicas (Particle, 2024).

Figura 4

Tarjeta Argon.



Nota: Tomada de Particle (2024)

La funcionalidad IoT que integra el Argon permite una comunicación continua entre el dispositivo y una estación de monitoreo local o remota, como un computador. Esta comunicación efectiva asegura que los datos biométricos sean enviados directamente al sistema en la nube, donde pueden ser almacenados, analizados y visualizados mediante interfaces interactivas, facilitando un monitoreo constante y proactivo del estado de salud del paciente. En definitiva, la combinación de la tarjeta con plataformas IoT, bases de datos en la nube e interfaces visuales proporciona una solución de bajo costo para la visualización remota de datos de salud u otra información. Esta tecnología es especialmente valiosa en zonas rurales o con limitada infraestructura médica, donde permite extender la cobertura de vigilancia clínica sin necesidad de instalaciones complejas o costosas.

2.1.11 Internet de las Cosas (IoT)

El IoT se ha consolidado como la tecnología clave para realizar la transformación digital de diversos sectores. Su principio fundamental radica en la interconexión de dispositivos físicos equipados con sensores, software y conectividad de red, permitiendo el almacenamiento, el análisis y transmisión de información.(docline, 2024).

El internet de las cosas tiene algunos beneficios y requerimientos necesarios para poder integrar esta tecnología las cuales son: (Mutual Médica, 2024).

- **Comunicación** los dispositivos deben tener comunicación wifi o bluetooth para poder integrarse a esta tecnología.
- **Conectividad global** los dispositivos se conectan entre sí e intercambian información a través del internet.

- **Análisis en tiempo real** los datos recopilados son analizados en tiempo real por algoritmos o por redes neuronales.
- **Automatización** permite el control automático a distancia de procesos casi en tiempo real.

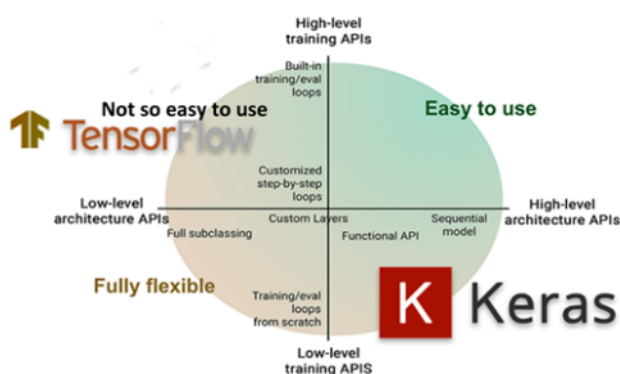
Los beneficios que ofrece el IoT, permite la implementación en el monitoreo de señales biomédicas, con el propósito de desarrollar un sistema de supervisión inteligente que optimice la atención médica y promueva la detección temprana de alguna anomalía del cuerpo. (Lun, 2019).

2.1.12 Red neuronal Keras

Es una API de red neuronales desarrollado en lenguaje de Python por Francois Chollet en 2015, pero en 2019 es adquirida por Google y es añadida a la biblioteca TensorFlow como una red neuronal de alto nivel de código abierto que se ejecuta sobre frameworks de TensorFlow, como se observa en la figura 5 como se clasifica la API en complejidad de programación y arquitectura. El objetivo es la creación de modelos neuronales sencillos rápidos y accesibles, pero también se puede crear arquitecturas complejas (Data Scientist, 2025).

Figura 5

Clasificación de la librería TensorFlow y Keras dentro las APIs.



Nota: Tomada de Particle (2024)

Dentro del aprendizaje profundo, una de las herramientas más utilizadas es la biblioteca *TensorFlow*, la cual permite la implementación de modelos de redes neuronales a través de su API de alto nivel *Keras* y sus principales características son (Dieckmann, 2023):

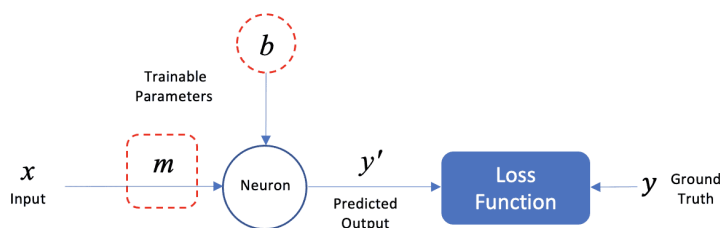
- Se integra en Google Colab lo que permite entrenar modelos de alta complejidad utilizando aceleración por GPU o TPU sin la necesidad de hardware especializado.

- Puede combinar módulos de capas neuronales con optimizadores, funciones de activación y esquemas de regularización para crear nuevos módulos.
- Es multiplataforma ya que admite al menos cinco motores de backend.
- Keras sacrifica la velocidad de procesamiento por la facilidad de uso por lo que se demora en el entrenamiento.
- La red no admite la creación de gráficos dinámicos.

Además del uso de redes neuronales profundas de Deep Learning como lo es esta red ayuda a modelar las condiciones adecuadas para cada necesidad, ya que además se debe aplicar la regresión lineal para organizar los datos y mejorar el ingreso de los datos a la red como se observa en la figura 6 el modelo de un red neuronal de Keras de una sola neurona la cual debe ser aplicada la regresión con toda la parte matemática que se encuentra detrás de la neurona (Dieckmann, 2023).

Figura 6

Modelo de red neuronal Keras de una neurona



Nota: Tomada de Learnopencv (2025)

2.1.13 Almacenamiento en la nube (Firebase)

Una de las bases de datos más conocida es Firebase la cual es desarrollada por Google, la cual tiene el mismo principio de una base de NoSQL con alojamiento en la nube permite almacenar datos desde cualquier plataforma que pueda ser configurada su comunicación con Firebase, su formato de almacenamiento es un archivo json, csv, xml entre otros, debido a su capacidad de manejar una gran cantidad de datos es una de más utilizadas por su fácil configuración y manejo, en su plataforma se tiene la posibilidad de configurar diferentes tipos de bases según la utilidad. La arquitectura de su funcionamiento se basa como se observa en la figura 7 (firebase, 2024).

- Herramientas para redactar, recopilar y adquirir información para la base.
- Backend se encarga de recibir solicitudes de escritura o lectura.
- La capa transportadora envía los mensajes a las plataformas o dispositivos del destino.

- Sdk muestra la notificación o se maneja el mensaje según el estado de la app en primer plano.

Figura 7

Descripción general de la arquitectura de Firebase



Nota: Tomada de Learnopencv (2025)

Las características que hacen de Firebase una de las más utilizadas en proyectos de minería de datos son las siguientes:

- Tiene una base llamada *Realtime Data base* que ofrece soporte para operaciones en tiempo real
- Los datos se almacenan en los servidores de Google.
- Permite la conexión de varias plataformas al mismo tiempo.
- Cuenta con base de datos Cloud con soporte para capacidades offline
- Su tráfico de datos es de baja latencia.
- Su sistema es multiplataforma para conexión con diferentes sistemas operativos.

Las características anteriormente mencionadas hacen que Firebase sea una base de datos segura ya que los datos alojados están protegidos, además que su comunicación es de baja latencia hace que esta base sea perfecta para el desarrollo de proyectos con IoT y más con la posibilidad de utilizar una Realtime para el desarrollo de este proyecto.

2.1.14 Ubidots

Es una plataforma en la nube que integra dispositivos con conexión para IoT gratuita, está especializada o diseñada para crear soluciones para monitorear y controlar equipos en forma remota. Se encuentra presente en varios sectores como la industria, el sector agrícola, medicina,

educación entre otros la plataforma tiene dos formas de accesos una que es gratuita como limitada configuración y creación de widget y otra de pago que tiene funciones de configuración personalizada y creación y configuración de dashboard ilimitado, sus principales características son las siguientes: (Espinosa Apolo, 2021)

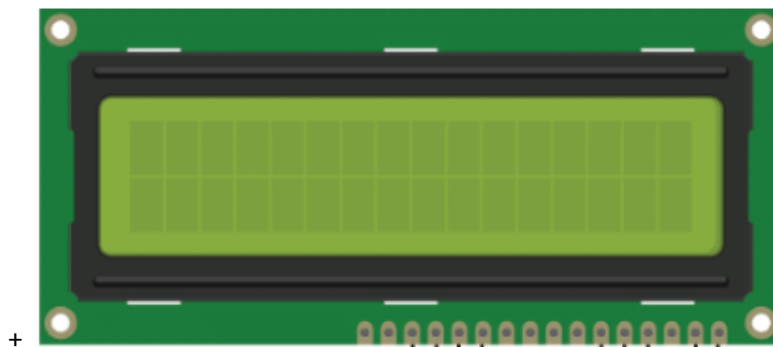
- Diseño por cuadros de mandos y de visualización fáciles de configurar.
- Modificación de la API en cualquier momento.
- Conexión de hardware con la nube y diversas plataformas con conexión IoT
- Conversión de datos del origen en información legible para la plataforma de forma automática.

2.1.15 LCD

Las pantallas lcd son indicadores electrónicos compuestos por cristal líquido como se observa en la figura 8, esta pantalla utiliza la luz del led que vienen incorporadas para crear caracteres al manipular las pequeñas celdas de cristal líquido, las cuales actúan como un filtro de color para permitir el paso de la luz y así poder formar pequeños pixeles que formaran letras o caracteres según este configurado el dispositivo, (Ali Moallin, 2017)

Figura 8

Pantalla LCD -.



Nota: Tomada de Moallin (2017)

Las ventajas de utilizar este tipo de pantallas en proyectos son las siguientes:

- **El costo** es muy rentable frente a otro tipo de pantallas como las OLED.
- **Eficiencia energética** es un dispositivo de bajo consumo energético.
- **Versatilidad** existen varios modelos y tamaños son óptimas para tarjetas existentes en el mercado

- **Multiplataforma** se puede decir que se conectan a todas las tarjetas de control existentes solo con una librería para configurar.

2.1.16 Batería de litio

Una batería de litio es un dispositivo electrónico activo que consta de dos, tres o más celdas de energía que se conectan en serie o en paralelo, la conexión de las celdas depende del voltaje y la corriente de la batería que se desee, como se sabe este tipo de baterías no requieren ningún tipo de mantenimiento preventivo o cuidado, por lo que su vida útil es larga siempre y cuando se siga las recomendaciones del fabricante para uso. Este tipo de baterías tiene varios modelos como es la batería tipo pouch observada en la figura 9 es más delgada, flexible y ligera.

Figura 9

Batería de litio.



Nota: Tomada de (Particle, 2024)

Las principales características de esta batería que hace recomendable para uso de equipos portátiles son las siguientes:

- Su alta densidad de energía.
- Su capacidad de almacenar grandes cantidades de energía en un tamaño compacto.
- Se tiene baterías de alto voltaje y bajo voltaje.
- Su larga vida útil.
- Carga rápida y baja descarga sin ausencia de efecto de memoria.

2.2 Descripción de la propuesta

La propuesta planteada en este proyecto es el diseño e implementación de un prototipo ambulatorio para el monitoreo y análisis de los signos vitales en tiempo real, mediante la integración de tecnologías IoT, almacenamiento en la nube y una inteligencia artificial. El controlador escogido es una tarjeta Argon Particle como nodo principal de adquisición, en la que se conectan los sensores

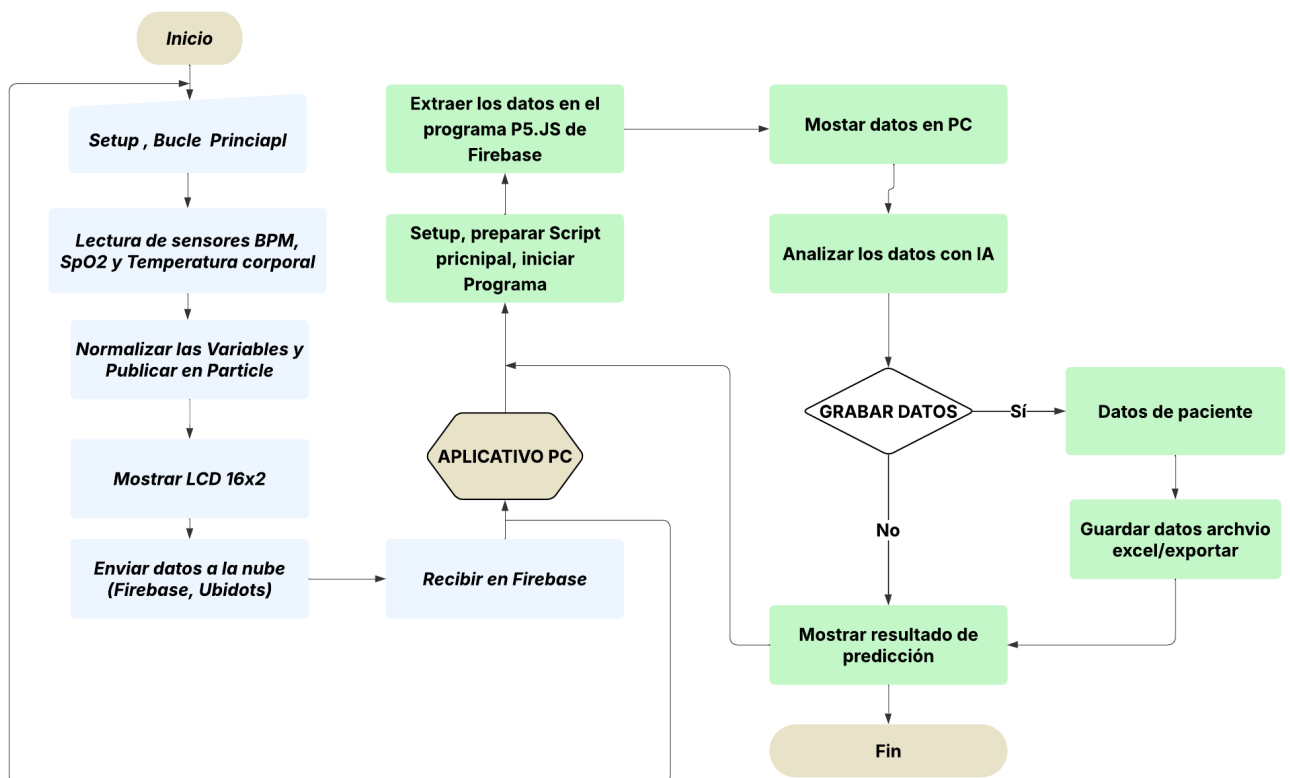
biomédicos MAX30102 y de temperatura, la información es enviada a una base de datos de Firebase Realtime Database, Posteriormente, un modelo de red neuronal procesa la información y predice posibles anomalías. Los resultados se visualizan mediante una interfaz desarrollada para una PC. De esta forma el sistema trata de mejorar la detección temprana de riesgos y fortalece las estrategias de telemedicina, ofreciendo una solución accesible para mejorar la atención hospitalaria.

a. Estructura general

Para entender el funcionamiento del proyecto se realiza el siguiente diagrama de bloques de la figura 4, la primera que es la recolección de las señales del individuo con los sensores, normalizar las variables y obtener la medida correcta de cada una de ellas ahí enviarlas con la comunicación IoT a Firebase y luego en el computador llega las señales al programa desarrollado que contiene la red neuronal entrenada lista para predecir alguna anomalía y una vez cumplido este ciclo se repite hasta cuando sea necesario.

Figura 10

Diagrama del sistema de monitoreo y análisis.



El prototipo desarrollado está compuesto por los principales componentes tanto de hardware como software que hacen posible la implementación del mismo:

1. **Tarjeta Argon Particle:** Es la que realiza el control central del sistema, encargado de recibir las señales de los sensores conectados a la tarjeta, procesa los datos iniciales y transmite hacia la nube o base de datos.
2. **Sensores biomédicos (MAX30102 y LM35/DS18B20):** Permiten la adquisición continua de parámetros como frecuencia cardíaca (BPM), saturación del oxígeno en la sangre (SpO2) y temperatura corporal, son dispositivos no invasivos.
3. **Base de datos en la nube:** Almacena de forma inmediata los datos adquiridos, garantizando disponibilidad y accesibilidad para su análisis posterior.
4. **Red neuronal profunda (TensorFlow/Keras):** Procesa los datos en un modelo previamente entrenado en Colab, identifica patrones anómalos y predice condiciones de riesgo que se puede dar en una persona.
5. **Aplicación de PC para monitoreo remoto:** presenta la mediada de las tres variables en tiempo real y la predicción de la red neuronal que se hace a los datos.
6. **Comunicación inalámbrica (Wi-Fi):** Establece la transmisión de datos desde la tarjeta Argon hacia la nube, permitiendo un monitoreo remoto en la aplicación desde cualquier PC.
7. **Integración con sistemas de telemedicina:** Posibilita la transmisión de los datos procesados hacia plataformas médicas o aplicaciones móviles, favoreciendo a los doctores.

b. Explicación del aporte

Funcionalidad del prototipo

El sistema de monitoreo de signos vitales propuesto en este proyecto se fundamenta en un modelo interactivo que permite la supervisión continua de la salud en tiempo real, tanto en entornos clínicos como domiciliarios, además contiene una red neuronal que permite procesar y analizar datos fisiológicos de forma automática que facilita la interpretación de la información. Esto lo convierte en

una solución accesible y práctica que fortalece la prevención y el diagnóstico temprano de anomalías en pacientes.

Recursos utilizados

Cada uno de los componentes utilizados desempeña un papel esencial en el correcto funcionamiento del sistema:

- Tarjeta Argon Particle actúa como unidad de adquisición y transmisión de datos hacia la nube en tiempo real.
- Sensor MAX 30102: Permite la medición de frecuencia cardíaca y saturación del oxígeno de manera precisa y no invasiva.
- Sensor de temperatura LM35/DS18B20: Registra la temperatura corporal con alta sensibilidad, complementando el monitoreo.
- Firebase Realtime Data base: Almacena los datos en la nube y asegura sincronización inmediata para su análisis.
- Red neuronal: Procesa las variables y predice posibles anomalías según llegan los datos.
- Interfaz gráfica: Representa de manera dinámica los valores y predicciones, accesible desde cualquier computador que tenga instalado la aplicación.
- Comunicación inalámbrica (Wi-Fi): Permite el flujo de datos continuo entre sensores, nube y visualización sin necesidad de cableado.

Actividades de evaluación del sistema

Para validar la efectividad del prototipo desarrollado, se plantean distintas pruebas de laboratorio ya que no se puede realizar pruebas en una casa de salud porque se debe cumplir con los artículos del MSP (Anexo 6) por lo que se optó por pruebas de laboratorio controladas con voluntarios:

- Pruebas de precisión de los sensores: Comparación de los datos obtenidos con dispositivos médicos como oxímetros y termómetros.
- Validación del modelo neuronal: Cálculo de métricas como precisión, sensibilidad y especificidad para analizar la capacidad de detección de anomalías y el criterio médico de una persona.
- Pruebas de usabilidad: Retroalimentación de un doctor general y los voluntarios sobre la facilidad de interpretación de los gráficos y alertas generadas.

- Alteración de variables: Ensayos con valores anómalos para evaluar la respuesta del prototipo y la robustez del algoritmo.

Desarrollo del prototipo

El desarrollo de este prototipo no solo contribuye a la digitalización del monitoreo de señales médicas, sino que también permite comprender la integración de hardware y software con modelos de inteligencia artificial. La implementación del prototipo y la programación de la tarjeta para los sensores que se eligieron conforman la primera etapa, el manejo de bases de datos en la nube es la segunda etapa y finalmente se tiene el diseño del algoritmo predictivo mediante redes neuronales y la visualización se desarrolla en una PC por lo que todo esto ofrece una perspectiva sobre impacto de la salud digital para la prevención de emergencias médicas y la optimización de recursos hospitalarios.

c. Estrategias y/o técnicas

El desarrollo del prototipo para el monitoreo de signos vitales se basó en un enfoque experimental y aplicado, orientado a validar la viabilidad técnica y la precisión de los resultados obtenidos. Para ello, se implementaron estrategias metodológicas que aseguran la confiabilidad, eficiencia y escalabilidad del equipo implementado:

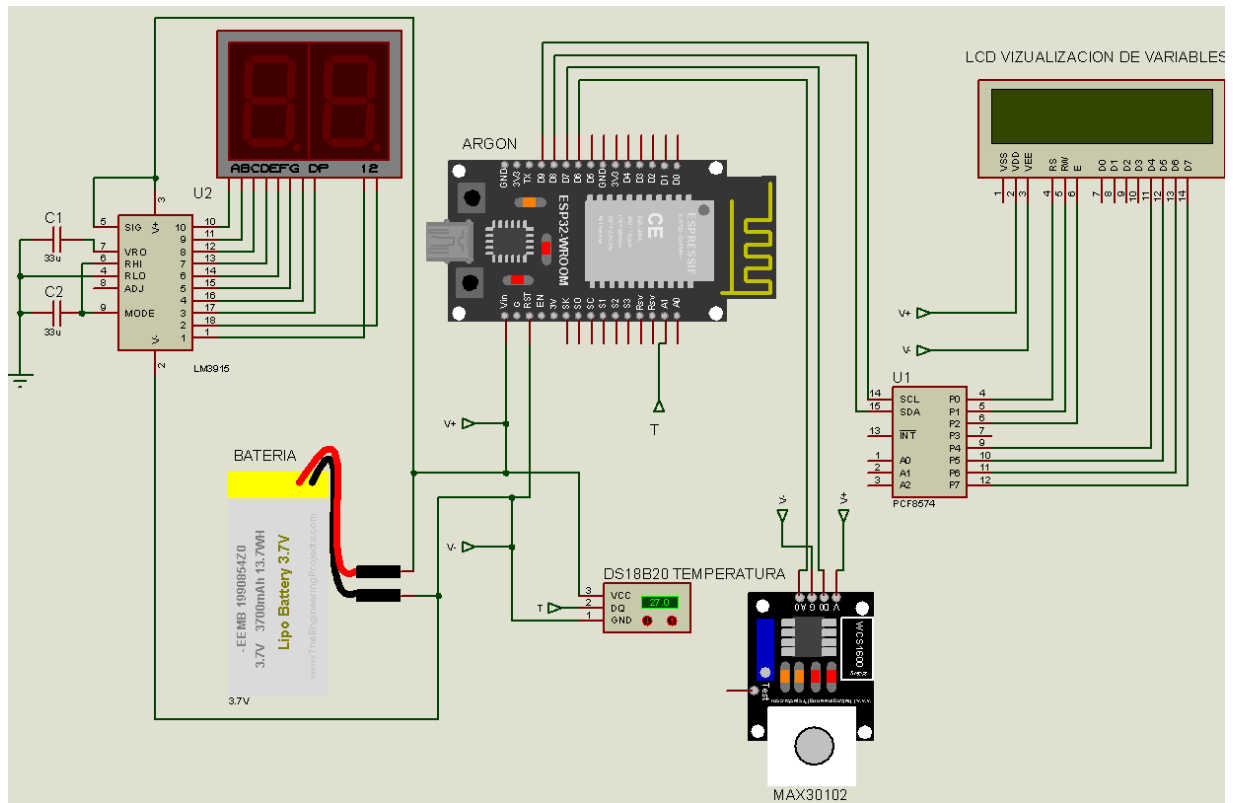
2.2.1 Desarrollo del prototipo

El prototipo desarrollado en esta investigación consta de una tarjeta Argon con los sensores anteriormente seleccionados para adquisición de las variables de cada uno de los pacientes, además consta de una pantalla lcd para la visualización de las variables localmente, todo esto se puede observar en el siguiente diagrama esquemático de la figura 11 en el cual se identifica los siguientes elementos:

- Pantalla Lcd con conexión I2C
- Display para conocer el nivel de batería
- Sensores max3010 y BS18B20/LM36
- Batería de litio de 3.7V con 3700 mah

Figura 11

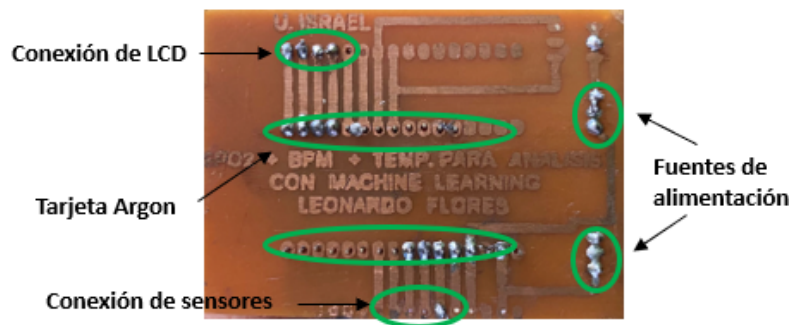
Diagrama esquemático del prototipo



Con el diagrama esquemático finalizado se tiene todas las conexiones a realizar se elabora una placa de circuito impreso en baquelita para el dispositivo como se observa en la *figura 12*, se puede observar que se monta la tarjeta Argon en la placa y la batería es al ser de 3.7V con 3700 mah es un poco grande así que todos los elementos están unidos por cables a la placa.

Figura 12

Circuito impreso del prototipo



El prototipo diseñado es alimentado por una batería de litio para lo cual se lo puso en una carcasa debido al tamaño de la batería en cual se integra todo el sistema como se observa en la

figura 13, se tiene el prototipo diseñado ya implementado y listo para ser programado, el equipo cuenta un lcd en cual se indica las variables medidas de las señales biomédicas además de indicador del porcentaje de la batería restante y sobresalen los cables a los cuales están conectados los sensores.

Figura 13

Equipo desarrollado del proyecto



2.2.2 Plataforma de configuración de tarjeta Argon

La configuración de la tarjeta de Argon se lo realiza en la plataforma llamada particle web IDE. A través de esta plataforma se configuró librerías para la programación de la lectura del sensor MAX 30102 y ADC para el sensor de temperatura y la definición de pines de lectura y escritura para el lcd, y todas las configuraciones necesarias para la conexión de la tarjeta a las diferentes plataformas mediante IoT.

2.2.2.1 Consola de particle web IDE

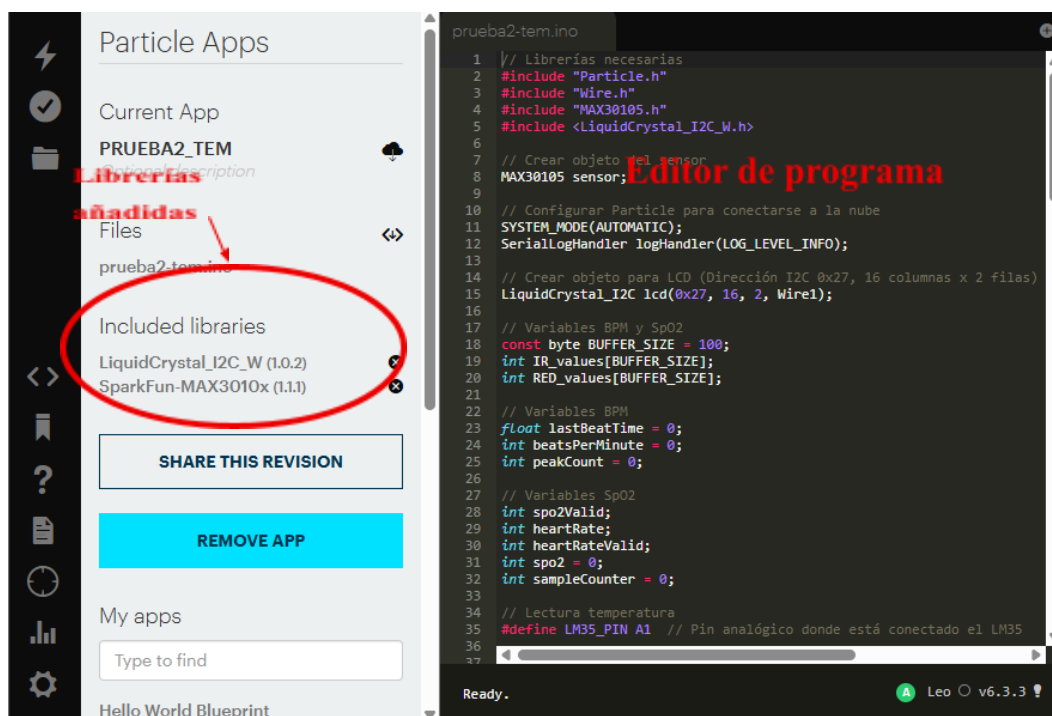
La interfaz de inicial de la plataforma indica las librerías cargadas necesarias para el MAX, lcd y comunicación con la nube como se muestra en la siguiente figura 14, el entorno de la plataforma y las configuraciones realizadas para el desarrollo del programa para lo cual se tiene lo siguiente.

Particle Apps: en esta sección se cargan todas las librerías y aplicaciones necesarias para el programa, además muestra el proyecto en el que se está trabajando, cargar programa, configuraciones, compilar y cargar el código.

Editor de programa: se desarrolla toda la programación en lenguaje C muy parecido al de Arduino en algunos casos con las mismas instrucciones, aquí se añaden las librerías cargadas.

Figura 14

Plataforma particle web IDE



2.2.2.2 Barra de herramientas web IDE

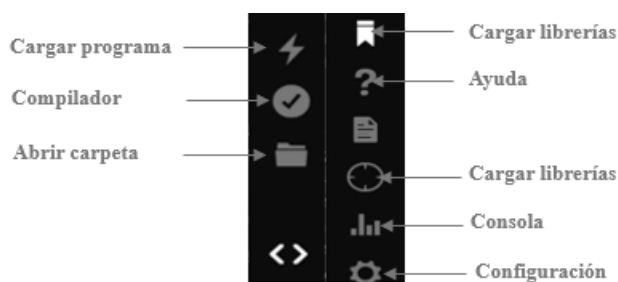
La barra de herramientas de la plataforma web IDE permite acceso a varias funciones de particle para la configuración y programación de la tarjeta con se observa en la figura 15, a continuación, se detalla cada icono de la barra de herramientas:

- Cargar programa: Envía el código escrito en el editor directamente a la tarjeta Particle (ej. Argon, Photon, Boron) sin necesidad de estar conectado con cables.
- Compilador: Compila el programa en la nube de Particle para verificar que no tenga errores antes de cargarlo y poder corregir los mismos.
- Abrir carpeta: Permite gestionar tus proyectos: abrir, crear o seleccionar trabajos ya realizados o reutilizar sketch en programas.

- Cargar librerías: Acceso al catálogo de librerías de Particle en la que se puede incluir al proyecto las necesarias, ejemplo sensores, comunicación, LCD.
- Ayuda: Brinda acceso rápido a las guías y soporte oficial de Particle.
- Consola: Abre la consola de Particle, donde puedes monitorear los datos enviados por el serial o eventos en tiempo real.
- Configuración: Ajustes del entorno Web IDE, como cuenta, dispositivos vinculados o preferencias de compilación.

Figura 15

Barra de herramientas de particle IDE

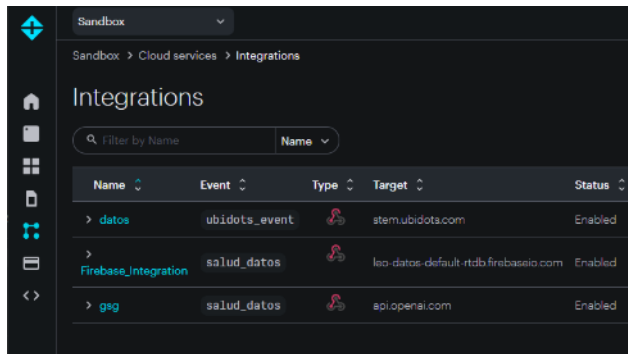


2.2.2.3 Integraciones IoT de plataforma particle

La plataforma Particle ofrece una integración nativa para proyectos IoT mediante el uso de tarjetas de la misma compañía como es Argon, las cuales permiten la conexión inalámbrica a internet a través de Wi-Fi, LTE o Bluetooth. Estas tarjetas facilitan la adquisición de datos desde sensores y enviarlos a la nube, como Firebase, Ubidots entre otras, a través de protocolos seguros como es el Webhooks como se observa en la figura 16. Una de sus principales ventajas es el Particle Cloud, que gestiona la comunicación bidireccional entre el dispositivo y plataformas o aplicaciones externas al mismo tiempo.

Figura 16

Integraciones IoT de particle



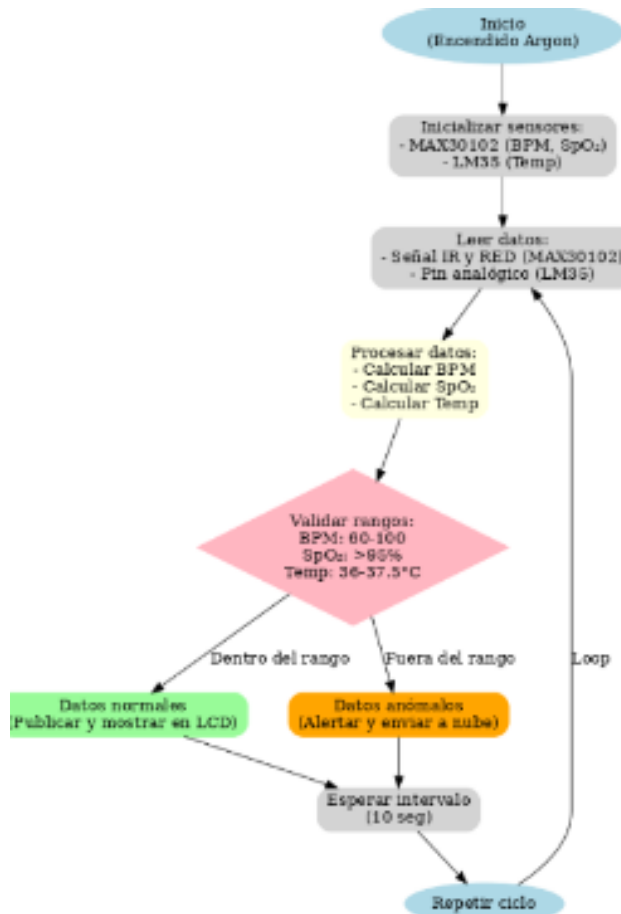
Los webhooks en Particle permiten establecer una comunicación directa entre el dispositivo IoT y servicios externos en la nube, sin necesidad de programar un servidor intermedio.

2.2.2 Programación de tarjeta Argon

En el programa de la tarjeta Argon se desarrolló toda la arquitectura para adquirir la señal del sensor MAX30102 que va conectado en el dedo índice de la izquierda del individuo, además se configuró el ADC para 12 bits para la lectura del sensor de temperatura se tener en cuenta que la fórmula para esta variable depende del voltaje en este caso es de 3.3 Vdc que se obtiene de la tarjeta, a continuación, se observa en la figura 17 el diagrama de flujo de la programación realizada en la tarjeta.

Figura 17

Diagrama de flujo de adquisición de señales de sensores



En la figura 18 se observa parte de la programación que desarrolla para la adquisición de las señales de los sensores y las librerías instaladas para todas las configuraciones de software necesario para la etapa de lectura y normalización de las variables antes de enviarlas a la nube.

Figura 18

Librerías cargadas al programa

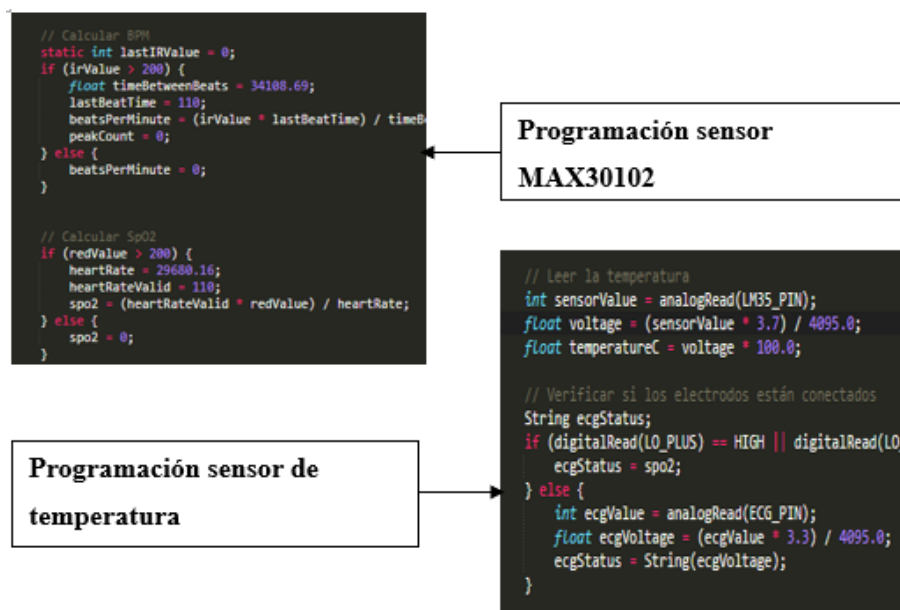
<pre> prueba2-tem.ino 1 // Librerías necesarias 2 #include "Particle.h" 3 #include "Wire.h" 4 #include "MAX30105.h" 5 #include <LiquidCrystal_I2C_W.h> 6 7 // Crear objeto del sensor 8 MAX30105 sensor; 9 10 // Configurar Particle para conectarse a la nube 11 SYSTEM_MODE(AUTOMATIC); 12 SerialLogHandler logHandler(LOG_LEVEL_INFO); 13 </pre>	Librerías cargadas para programar sensores y pantalla lcd
---	---

La configuración para la lectura de los sensores es observado en la figura 19, este código es depurado al realizar pruebas de funcionamiento para verificar los datos con dos dispositivos

verificados para realizar una comparación de las medidas con nuestro prototipo y de esta manera corregir el código fuente hasta que el error sea mínimo con respecto al oxímetro y al smartwatch , se realizó la corrección en el código y en la librería se modifica la velocidad de muestreo para tener un valor de lectura muy cercano al real con respecto a los dispositivos.

Figura 19

Código de lectura de sensores



2.2.3 Comunicación con Firebase y Ubidots

2.2.3.1 Configuración de Webhook para Firebase y ubidots

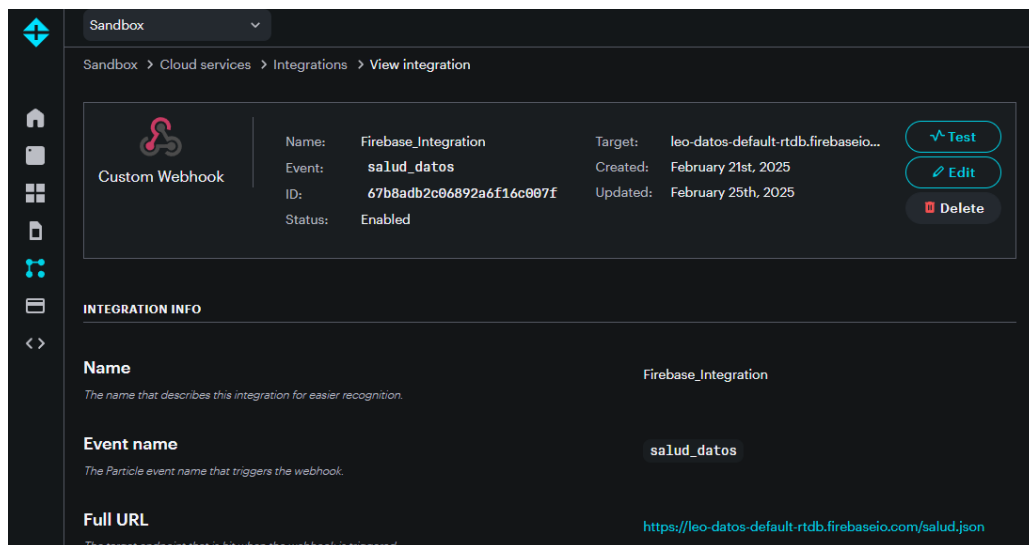
Una vez saneado y verificadas las medidas de los sensores se procedió a realizar la conexión de IoT con la tarjeta Argon para lo cual se utiliza la herramienta llamada Webhook porque permite conectar bidireccionalmente a las diferentes plataformas porque el envío de datos o información es continua. En la figura 20 se observa la configuración de la plataforma IoT con una base de datos de Firebase.

Webhook: configurado para comunicarse con un base de datos “Realtime” de Firebase se debe tener creada una cuenta para poder configurar la conexión.

Serial: se debe configurar el serial de la tarjeta para que envíe los datos a la nube y pueda enviar datos al webhook.

Figura 20

Configuración para comunicación con Firebase



En la consola se debe realizar las siguientes líneas de código como se observa en la figura 21 para poder utilizar la conexión creada para lo cual se debe configurar lo siguiente:

System mode: define el modo de conexión de la tarjeta con la nube se debe configurar en modo automático.

Serial: abre el puerto serial para la comunicación y habilita el envío de mensajes a las diferentes plataformas de integración en este caso a Firebase y Ubidots.

Funciones: es la encargada de enviar datos a las diferentes plataformas respectivamente como se observa en la figura 21, una vez configurada el webhook y la tarjeta se puede probar la comunicación.

Figura 21

Configuración para comunicación con Firebase y ubidots

```
// Configurar Particle para conectarse a la nube
SYSTEM_MODE(AUTOMATIC);
SerialLogHandler logHandler(LOG_LEVEL_INFO);

// **Enviar datos a Firebase y Ubidots**
enviarDatos(beatsPerMinute, spo2, temperatureC, ecgStatus); // Enviar a Firebase
enviarDatosUbidots(beatsPerMinute, spo2, temperatureC); // Enviar a Ubidots
```

Los datos son almacenados en un archivo llamado “salud datos”, el tiempo de almacenamiento de cada dato depende de la tasa de muestreo con la que la tarjeta adquiere las señales la cual está configurada cada 5sg.

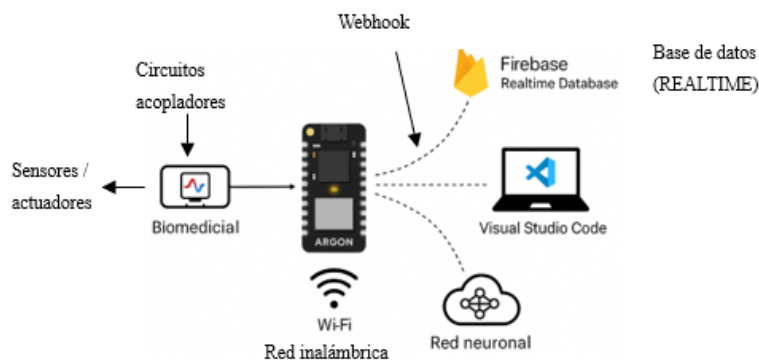
Una vez realizada la base de datos está disponible para ser utilizada en cualquier aplicación por lo que llevamos los datos a través de p5js al computador o cualquier dispositivo que contenga el programa sin importar la ubicación.

2.2.3.2 Configuración de la base de datos

La integración con la nube se hace con las herramientas que permite la tarjeta Argon por lo que no es necesario integrar otro tipo de tarjeta. La plataforma utiliza el protocolo llamado “coat” para la transmisión de información en tiempo real desde la tarjeta, proporcionando una comunicación bidireccional en caso de que se requiera se puede agregar diferentes aplicaciones al mismo tiempo sin perder la fiabilidad ya que es estable de baja latencia como se observa el flujo de datos puede ser como en la figura 22.

Figura 22

Diagrama de comunicación con la nube



Realtime data base se escoge esta base por su robustez y capacidad para gestionar grandes volúmenes de información en tiempo real, lo que permite un control preciso y personalizado, ya que guarda los datos en la base y automáticamente están disponibles para ser utilizados o leídos desde otra plataforma, para lograr esto se debe configurar la base de la siguiente manera como se observa en la figura 23.

Figura 23

Configuración de base de datos en Firebase

```
// Import the functions you need from the SDKs you need
import { initializeApp } from "firebase/app";
import { getAnalytics } from "firebase/analytics";
// TODO: Add SDKs for Firebase products that you want to use
// https://firebase.google.com/docs/web/setup#available-libraries

// Your web app's Firebase configuration
// For Firebase JS SDK v7.20.0 and later, measurementId is optional
const firebaseConfig = {
  apiKey: "AIzaSyCg_Wwo0usSfnRm7J-Bn_kEpPsPZQuJpxs",
  authDomain: "leo-datos.firebaseio.com",
  databaseURL: "https://leo-datos-default-rtdb.firebaseio.com",
  projectId: "leo-datos",
  storageBucket: "leo-datos.firebaseio.com",
  messagingSenderId: "771433349795",
  appId: "1:771433349795:web:09ff7695b2cea19e7a3516",
  measurementId: "G-PS5T1LQDFQ"
};

// Initialize Firebase
const app = initializeApp(firebaseConfig);
const analytics = getAnalytics(app);
```

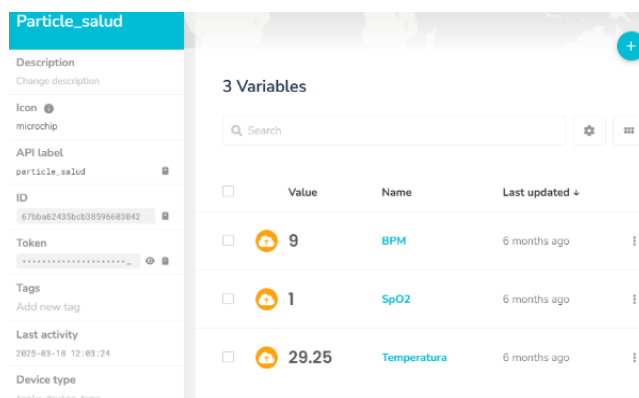
Los datos provenientes de la tarjeta son guardados en archivo JSON, por lo que para procesarlos y enviarlos a la aplicación donde se encuentra la red y la interfaz de usuario es muy fácil, en la nube los datos se los guarda con etiquetas para luego ser leídas por el programa de VScode.

2.2.3.3 Configuración de Ubidots

En la plataforma ubidots se debe crear una cuenta de forma gratuita o llamada ubidots stem que su uso es gratuito si desea la versión original solo debe ir a la página de inicio y elegir algún plan de pago, pero este proceso no es necesario ya que estamos trabajando con una arquitectura cerrada. Una vez configurada una cuenta se procede a crear las tres variables que tiene el sistema para poder graficar y crear los histogramas que deben coincidir con los datos guardados en Excel que también pueden ser graficados. figura 24.

Figura 24

Dashboard de las variables en ubidots



En la plataforma de ubidots se configuro las tres variables de forma individual, pero si es necesario se puede configurar en sola gráfica, en este caso no es necesario en la tarjeta también hay que configurar el puerto para para el webhook creado empiece a enviar información esto se lo hace con el código mostrado en la figura 21.

2.2.4 Red neuronal Keras

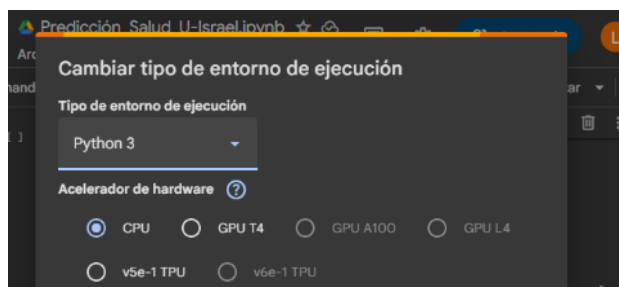
La red neuronal desarrollada para el trabajo es entrenada en la plataforma de colab, se utilizó Keras con las bibliotecas de TensorFlow, para la programación de la red primero hay que cargar las librerías a colab para poder realizar el script en Python que permite cargar y explorar el conjunto de datos recopilado. Este data set, compuesto por 200,000 registros, incluía observaciones tanto de personas sanas como de personas con condiciones fisiológicas anómalas para programar se utiliza el siguiente proceso:

1. Configurar entorno

En la pantalla principal de Colab se debe seleccionar el tipo de hardware puede ser GPU o TPU lo que se logra con esta configuración inicial es acelerar mucho el entrenamiento de redes neuronales sin generar pérdidas o que afecte a la red creada, en la figura 25 se observa el tipo de configuración a utilizar.

Figura 25

Dashboard de las variables en ubidots



2. Importar librerías necesarias

Se debe importar todas las librerías de Tensor Flow/Keras para la red neuronal que se va desarrollar debido a que se está programando la red desde cero las librerías que se cargan se pueden observar en la figura 26, por lo que para el desarrollo se necesita librerías para:

- Manejo de datos la más conocida numpy, pandas y para la visualización la que se utiliza es Matplotlib.
- Las librerías de Keras y Tensor Flow para la red, métricas y reportes.
- Librería para la regresión lineal.

Figura 26

Dashboard de las variables en ubidots

```
# Importar Librerías
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report
import tensorflow as tf
from tensorflow import keras
```

3. Cargar los datos

Una vez configuradas y cargadas todas las librerías necesarias para desarrollar el programa, se debe crear una base de datos con la información que será utilizada para el entrenamiento de la red se puede cargar un dataset desde:

- Archivos locales subida manual desde el computador este archivo por lo general es json o csv, como es el caso se hizo una recolección de datos con el prototipo para entrenar la red.
- Google Drive son archivos alojados en la nube que también se pueden cargar.
- Fuentes públicas que tienen datos reales como Kaggle o datasets de Tensor Flow.

4. Preprocesar los datos

La información que se sube al Colad debe ser ordenada clasificada para que la red pueda entender los datos que se ingresa, como se sabe estamos ingresando tres variables numéricas que son las medidas de los sensores biomédicos que alrededor se tiene un conjunto de 200000 datos aproximadamente por lo que se debe hacer el siguiente proceso.

- Primero se debe eliminar valores nulos o erróneos con el código `df = df.dropna()`.
- Codificación de los datos es decir se debe transformar las etiquetas en categorías numéricas si es clasificación.

- Lo más importante es separar los datos en tres grupos que son entrenamiento, validación y prueba eso se lo puede hacer en dos matrices.

5. Definir la arquitectura de la red neuronal

El proceso matemático que engloba la red Keras creada para el análisis de las señales biomédicas es muy sencillo de entender cómo se detalla a continuación la matriz de entrada consta de 3 neuronas por las tres variables

$X = [BPMS\ SpO2\ Temperatura] = \text{matriz de 3 neuronas por 3 variables de entrada}$

Para el cálculo de la capa oculta de la red que permite las combinaciones de la variable se tiene la siguiente matriz, cabe recordar se utiliza capas ocultas con Relu las cuales son escogidas luego de la investigación ya que son las adecuada para el tipo de trabajo, por lo que se tiene lo siguiente operación matemática para cada capa es:

$$\sum_{i=0}^l Z = W(i) \cdot X(i-1) + b(i) \quad \text{Cálculo de capa por capa}$$

$a = f(z)$ función de activación de la red

Donde:

- W es la matriz de pesos de la neurona.
- b es un vector de sesgo para la salida.
- La función f(z) es la que hace referencia a la activación ReLu.

El modelo tiene dos capas ocultas para refinar los datos como se sabe la primera capa oculta nos ayuda a revisar las combinaciones de la entrada con 16 neuronas luego del resultado pasamos por una segunda capa oculta para eliminar falsos positivos refinar la predicción de la red para lo cual tenemos la siguiente función,

$$\sum_i^n f(z) = \max(0, z)_i ; \text{Segunda capa para eliminar falsos positivos}$$

Las redes creadas en Keras tienen una capa de salida con las neuronas necesarias o iguales a las respuestas o predicciones que se va realizar para esta capa se usa softmax que transforma los valores devueltos en probabilidades para lo cual se tiene la siguiente matriz que se forma con la fórmula a continuación,

$$y_i = \frac{e^{z_i}}{\sum_j e^{z_j}} \text{ Capa de salida}$$

Donde

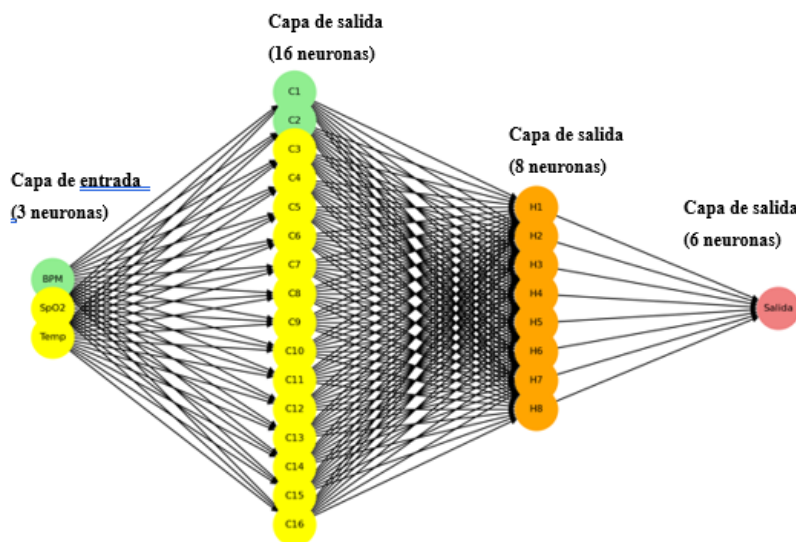
- y_i es la probabilidad que un paciente pertenezca a la clase i .
- K es el número de clases que se tiene en este caso 6 clases o seis posibles combinaciones de respuestas

Arquitectura de red neuronal: el tipo de red neuronal que se está construyendo es de tipo secuencial con 4 capas y 33 neuronas en total, distribuidas en la capa de entrada, dos capas ocultas y la capa de salida se escoge este tipo de red ya que es ideal para procesar datos numéricos como son las señales que se están adquiriendo. En la figura 27 se representa la red neuronal diseñada para el trabajo la cual consta de lo siguiente:

- **Capa de entrada** 3 neuronas una por cada variable de entrada.
- **Primera capa oculta** se utilizan 16 neuronas que son las combinaciones de las tres variables de entrada
- **Segunda capa oculta** se utilizan 8 neuronas para refinar las combinaciones y generar representaciones más abstractas.
- **Capa de salida** 6 neuronas que representan las salidas o anomalías que se puede tener.

Figura 27

Modelo de aprendizaje profundo secuencial de la red Keras



El modelo matemático que utiliza la red neuronal Keras es sencillo, además se lo compara con la regresión lineal para comparar las predicciones de la red que se está desarrollando para mejorar la precisión de la red, el código que se elabora en colab para crear la red es que se puede observar en la figura 28 como están distribuidas las capas con sus respectivas neuronas.

Figura 28

Programación de red neuronal para el aprendizaje profundo

```
# ♦ **Red Neuronal con Keras**
keras_model = keras.Sequential([
    keras.layers.Dense(16, activation='relu', input_shape=(input_dim,)),
    keras.layers.Dense(8, activation='relu'),
    keras.layers.Dense(len(df["Class"].unique()),
])
```

Para definir todo esto se utiliza la siguiente fórmula que ayuda entender la creación de la red neuronal la cual explica cómo es la arquitectura de la red en cada una de sus etapas

6. Configurar el modelo

Para definir la función de pérdida Seleccionar las métricas que, por entropía, debido a que lo hace con Keras se aplica una entropía cruzada como una función de pérdida es decir mide las probabilidades de predicciones de la red y las reales, matemáticamente la entropía cruzada de a la red se explica como:

$$L = - \sum_{i=1}^n y_i (\hat{y}_i)$$

Donde:

- $\hat{y}_i$ Es la probabilidad de predicciones de la red.
- y_i es la clase real codifica en la segunda capa

7. Entrenar el modelo

Con el modelo de red configurada y lista para entrenar se ejecuta el entrenamiento con los datos de, se debe especificar número de iteraciones que vamos a realizar con todos los datos en este caso escogimos 50, además se debe especificar la cantidad de muestras procesada a la vez. Esto se observa en la figura 29 se valida el rendimiento en el conjunto de validación.

Figura 29

Entropía de la red neuronal, iteraciones

```
keras_model.compile(optimizer='adam',  
                    loss='sparse_categorical_crossentropy',  
                    metrics=['accuracy'])  
history = keras_model.fit(X_train_scaled, y_train,  
                          epochs=50,  
                          batch_size=8,  
                          validation_data=(X_test_scaled, y_test))
```

8. Evaluar el modelo

- Se prueba con los datos de prueba (test set).
- Se obtienen métricas finales: precisión, error, matriz de confusión.

9. Visualizar resultados

Una vez entrenado el modelo y haber depurado la red neuronal es decir reduciendo al máximo los falsos positivos y tener una precisión cercana a 1, se puede guardar el modelo entrenado para usarlo en aplicaciones reales en formato reales como es el p5js. Ya que esta plataforma admite los archivos que contiene una red Keras entrenada lista para hacer predicciones de los datos que ingresen a la misma,

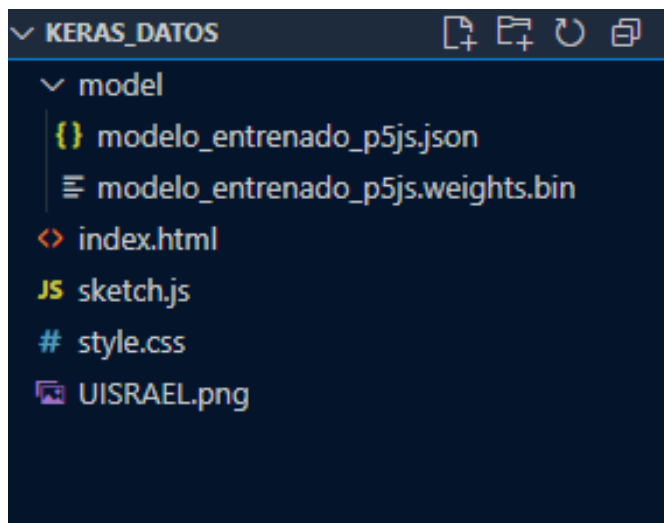
- Gráficas de pérdida y precisión durante las iteraciones de la red en el entrenamiento de la red.
- Comparar predicciones vs valores reales para verificar la red neuronal y exportar el modelo entrenado para poder utilizarlo en VScode para crear el aplicativo.

2.5.5 Desarrollo de aplicación para la PC

Sistema: el sistema es desarrollado en una la plataforma Visual Studio Code utilizando la herramienta de psj 5 para desarrollar el entorno de la aplicación que contenga la red neuronal previamente entrenada en colab para predecir este entorno de aplicación contienen las siguientes pestañas de programación en las cuales también se encuentra el modelo entrenado.

Figura 30

Archivos del programa desarrollado para la PC



Los archivos que conforman el aplicativo desarrollado para la interfaz del usuario en el computador son los siguientes:

- Index es el archivo que contiene toda la distribución de la interfaz es decir es el maquettador de la página.
- Sketch es donde se aloja toda la programación de la aplicación, ahí también se encuentra la red neuronal entrenada.
- Styles es donde se configuran todos los estilos de la interfaz como color, tipo de letras y formas.
- Model es la red Neuronal de Keras entrenada para el prototipo.

Interfaz de usuario (UI): La interfaz del usuario de la aplicación fue diseñada para ser muy intuitiva y amigable, en la cual se proporciona la visualización de datos en tiempo real y además tiene la opción de guardarlos localmente, lo cual servirá para crear historias clínicas por paciente, en la siguiente figura 31 se muestra la interfaz del usuario la cual será manejado por el médico o responsable.

Figura 31

Visualización de datos del dispositivo Particle Argon en el aplicativo.



Los resultados fueron visualizados en la interfaz muestra en tiempo real los datos recolectados del paciente, pero si se desea se tiene la opción de guardar la información por cada paciente se almacena en un Excel todos los datos del paciente se puede controlar desde qué momento se empieza a grabar hasta cuándo lo desee, esto ayuda a generar métricas, histogramas ya que el Excel extraído del programa contiene las variables de las tres variables y de esta manera complementar el proyecto y llevar datos ya no en papel y aplicando la transformación digital de los procesos en este caos una consulta médica de rutina .

2.3 Validación de la propuesta

La propuesta del prototipo de monitoreo remoto IoT y predicción de anomalías con IA, fue sometida a una evaluación de tres especialistas en desarrollo de software, automatización y domótica asegurando así la factibilidad y pertinencia técnica del sistema de monitoreo, para determinar su posible uso en casas de salud o futuros estudios a partir de este proyecto. Cada evaluador aplicó una rúbrica de evaluación ya definida en la que se analizaron criterios fundamentales para la validación del proyecto. Los documentos firmados por los evaluadores se incluyen en ANEXO 7

Tabla 4 *Perfil de validadores*

Nombres y Apellidos	Años de experiencia	Titulación Académica	Cargo
David R. Rivas L	18	Ph.D En Tecnologías de la información y Telecomunicaciones Master Universitario en redes de telecomunicaciones para los países en desarrollo Ingeniero en Electrónica e Instrumentación	Jefe de Comunicaciones de la Universidad de las Fuerzas Armadas ESPE-Latacunga
Gissela M. Guanoluisa P	5	Msc. En Ciencias de Datos Ingeniera en software	Docente Investigador en Departamento de seguridad y defensa Esforce.
Walter Villarroel H.	8	Ingeniero en Electrónica e Instrumentación	Gerente General de Vgoldtrack Docente en Instituto Cotopaxi

Tabla 5 *Escala de evaluación de Ph.D David Rivas*

CRITERIOS	EVALUACIÓN y SEGÚN la IMPORTANCIA Y REPRESENTATIVIDAD				
	En Total Desacuerdo	En Desacuerdo	Ni de Acuerdo Ni en Desacuerdo	De Acuerdo	Totalmente Acuerdo
Impacto					X
Aplicabilidad				X	
Conceptualización					X
Actualidad					X
Calidad Técnica					X
Factibilidad					X
Pertinencia					X

Tabla 6 Escala de evaluación Msc. Gissela Guanoluisa

CRITERIOS	EVALUACIÓN y SEGÚN la IMPORTANCIA Y REPRESENTATIVIDAD				
	En Total Desacuerdo	En Desacuerdo	Ni de Acuerdo Ni en Desacuerdo	De Acuerdo	Totalmente Acuerdo
Impacto					X
Aplicabilidad				X	
Conceptualización					X
Actualidad					X
Calidad Técnica					X
Factibilidad				X	
Pertinencia					X

Tabla 7 Escala de evaluación Ing. Walter Villarroel

CRITERIOS	EVALUACIÓN y SEGÚN la IMPORTANCIA Y REPRESENTATIVIDAD				
	En Total Desacuerdo	En Desacuerdo	Ni de Acuerdo Ni en Desacuerdo	De Acuerdo	Totalmente Acuerdo
Impacto				X	
Aplicabilidad					X
Conceptualización					
Actualidad				X	
Calidad Técnica				X	
Factibilidad					X
Pertinencia					X

2.4 Matriz de articulación de la propuesta

En la presente tabla 8 matriz se resume la articulación del prototipo realizado con el sustento teórico, metodológico, técnico y tecnológico utilizados para el desarrollo.

Tabla 8 *Matriz de articulación*

Ejes o partes principales del proyecto		Breve descripción de los resultados de cada parte	Sustento teórico que se aplicó en la construcción del proyecto	Metodologías, herramientas técnicas y tecnológicas que se emplearon
1	Definición de componentes y tecnologías	Selección de Tarjeta para conexión IoT y sensores no invasivos (temperatura, SpO2, BPM), además escoger la base de datos en la nube.	Internet de las Cosas Salud Digital Computación en la nube Normativa ecuatoriana de salud.	Investigación bibliográfica sobre arquitectura IoT para monitoreo médico, revisión de estándares de salud, y principios de bases de datos en tiempo real.
2	Elaboración de la comunicación con la nube	Elaboración del circuito con los sensores y tarjeta Argon Particle, configuración de lectura y envío de datos a Firebase, Backend para procesamiento de datos, red neuronal entrenada e implementación de interfaz de usuario	Redes Neuronales Artificiales Programación orientada a objetos Visualización en tiempo real Firebase SDK	Diseño modular. Uso de Visual Studio Code, Keras, Tensor Flow, Firebase, y p5.js. Arquitectura cliente-servidor y pruebas unitarias del flujo de datos.

3	Implementación experimental	Instalación de sensores a la placa, tarjeta y suministro de energía, evaluación de precisión de la IA, verificación de comunicación del sistema con Firebase	Machine Learning supervisado Telemedicina Evaluación basada en métricas (accuracy, recall) Arquitectura de sistemas distribuidos	Aplicación del modelo en fase experimental con base de datos simulada. Medición de tiempos de respuesta y precisión. Validación funcional de alertas médicas por IA.
4	Validación de resultados mediante pruebas de funcionamiento	Confirmación de la fiabilidad del prototipo mediante comparación de lecturas con otros dispositivos, evaluación de consumo energético y autonomía del prototipo.	Metodología de prueba y validación en prototipos de Medición y monitoreo	Pruebas con personas en estado de reposo y actividad física para asegurar la operatividad del sistema
5	Optimización y futuras mejoras	Se pueden hacer mejoras en la calibración de sensores al cambiar por otros de más alta fidelidad y precisión y la optimización de la red neuronal con mayor capacidad de procesamiento.	Optimización de la red neuronal con más tiempo de entrenamiento, Integración de nuevas tecnologías de sensores biomédicos, Evaluación de la escalabilidad del sistema.	Pruebas en hospitales, Integración con nuevos dispositivos IoT, Análisis de escalabilidad en entornos de difícil acceso.

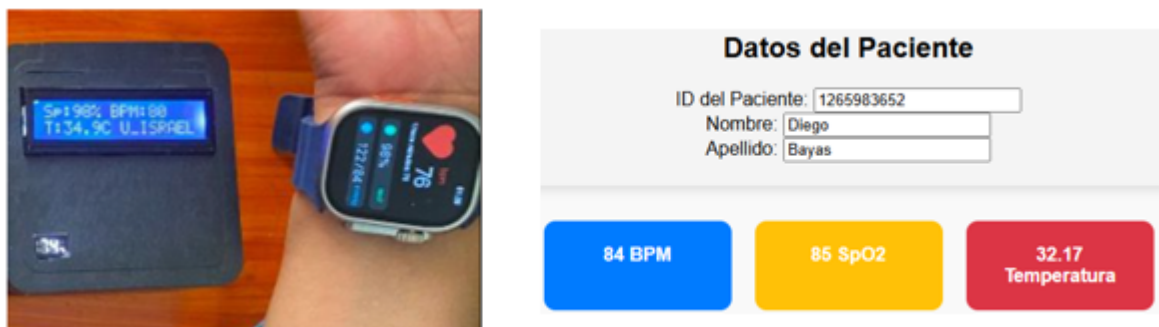
2.5 Análisis de resultados. Presentación y discusión

2.5.1 Análisis de resultados del sistema

Para validar el rendimiento del prototipo monitoreo desarrollado luego de la investigación previa, se realizó de la siguiente manera con una comparación entre los valores recolectados por los sensores conectados a la tarjeta Argon y los valores medidos por dispositivos médicos convencionales como el oxímetro y el termómetro, además con un reloj inteligente como se observa en la figura 32. Además, se grafica los signos vitales leídos en Ubidots y en Colab para verificar el resultado de las predicciones de tres pacientes de cual se hizo las pruebas controladas.

Figura 32

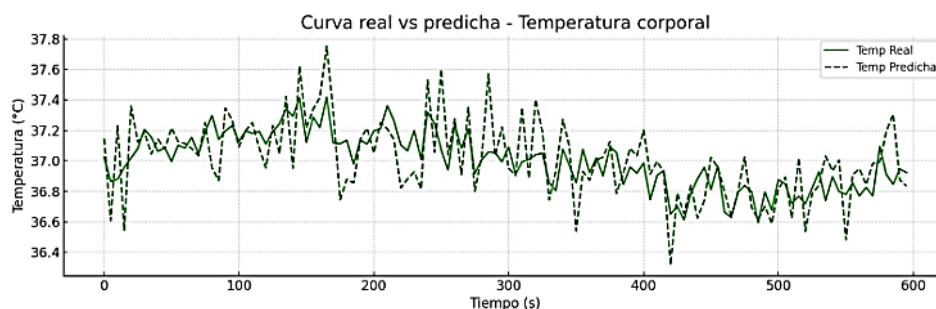
Muestras de sistema local y remoto.



La convergencia cercana entre la curva de entrenamiento y la de validación demuestra que no hubo sobreajuste, lo que sugiere un buen nivel de generalización del modelo hacia datos nuevos. Por otro lado, la gráfica de datos revela un SpO2 no tan normal hasta estabilizarse en torno al 96.5 % en ambas curvas, alcanzando una precisión sobresaliente todo esto en la figura 34 se observa una de las variables para este caso.

Figura 33

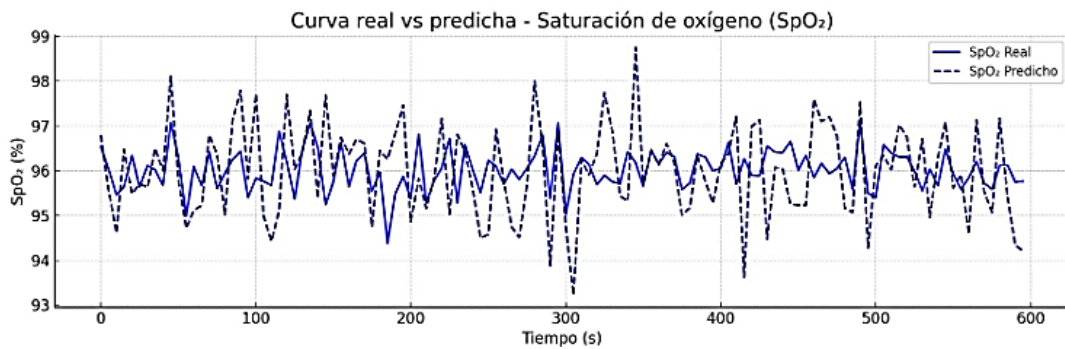
Curva de variable de temperatura medidas y predicha



Lo mismo para las variables de SpO2 se observa en la siguiente figura 34 los valores para la predicción del modelo creado.

Figura 34

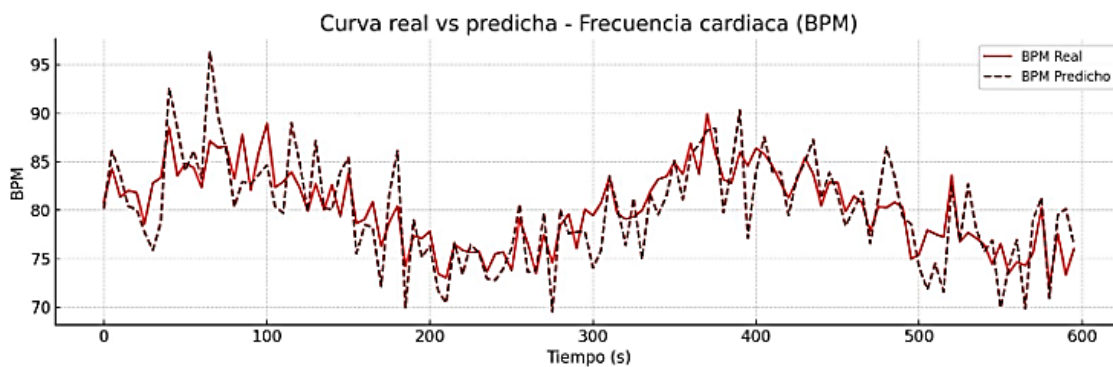
Curva de variable de SpO2 medidas y predicha



Finalmente, en la figura 35 se observa los valores para la variable de BPM por que se tiene las tres variables para la predicción la red

Figura 35

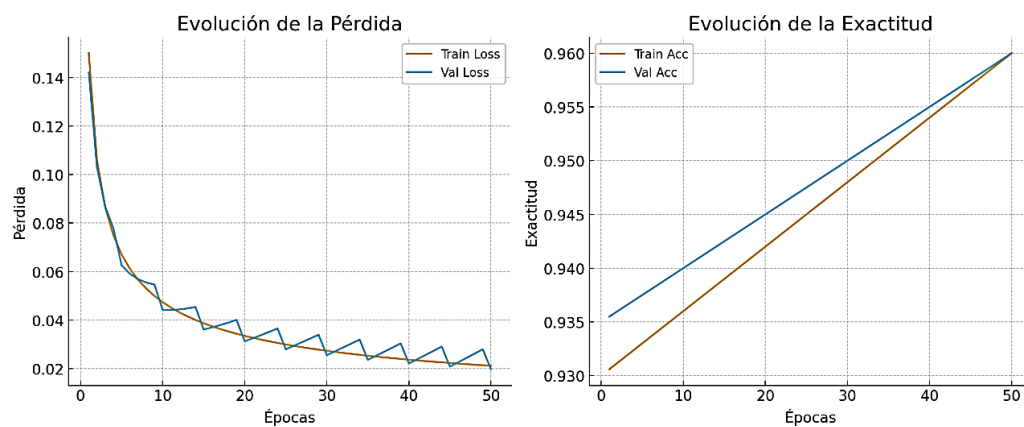
Curva de variable de BPM medidas y predicha



El comportamiento del modelo se resume en las gráficas de pérdida (loss) y exactitud (accuracy), tanto para los datos de entrenamiento como para los de validación. En la gráfica de pérdida, se observa una rápida disminución inicial que alcanza valores estables por debajo de 0.04, lo que indica que el modelo fue capaz de minimizar eficazmente el error en sus predicciones todo esto en la figura 36.

Figura 36

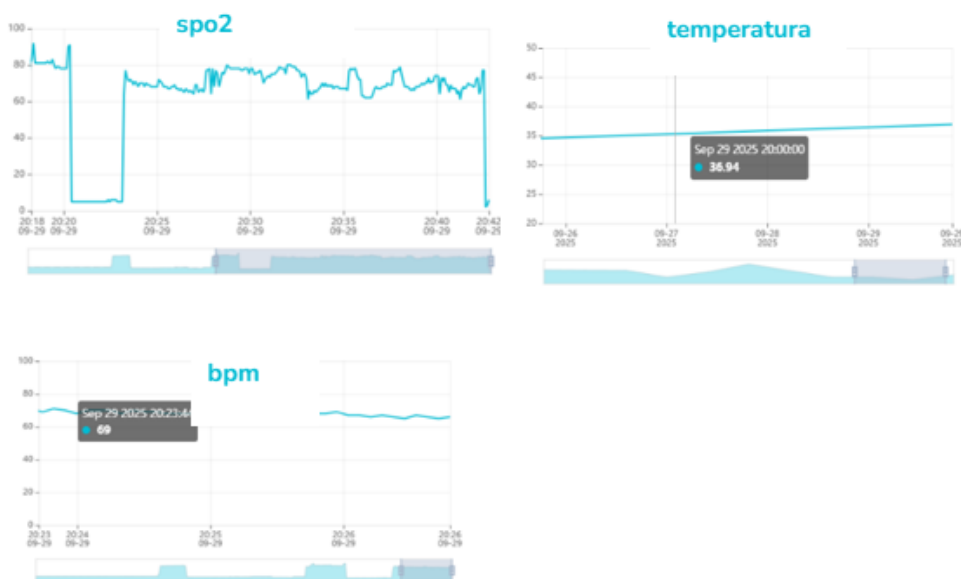
Exactitud y pérdida de red neuronal



Este comportamiento evidencia que la red neuronal logró aprender correctamente los patrones complejos presentes en los datos biomédicos, manteniendo un equilibrio entre aprendizaje y capacidad de predicción. En conjunto, estas gráficas validan la efectividad del modelo para ser utilizado en sistemas de monitoreo en tiempo real, proporcionando diagnósticos preliminares confiables a partir de datos fisiológicos del paciente por lo que se realiza una prueba de un paciente para verificar la predicción de la red como se observa en la figura 34 es la muestra de las tres variables y da una predicción de riesgo de hipoxia para conseguir estas señales el paciente hizo ejercicios para conseguir los datos.

Figura 37

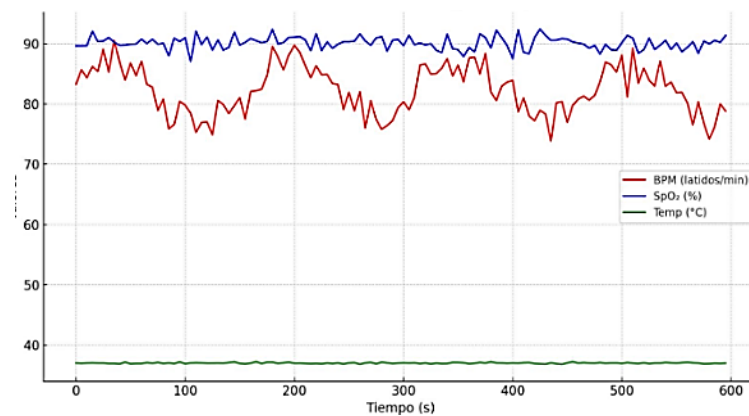
Gráficas de la red neuronal BPM de una paciente gráfica de Ubidots



La predicción de una anomalía que tenga las tres variables alteradas o una de la tres también se tiene la siguiente gráfica en referencia a las anterior de ubidots la gráfica generada por la red neuronal esto se observa en la figura 38 lo que ayuda a entender la regresión lineal para lograr la predicción de los datos

Figura 38

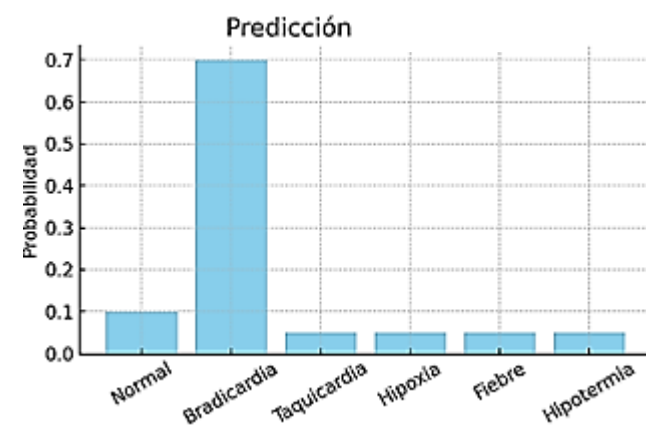
Probabilidad de la predicción paciente según la red



La predicción de la red de la muestra de una paciente observada anteriormente, se tiene la siguiente probabilidad de tener una anomalía según los datos para esta prueba al individuo se lo hizo cuando terminó una actividad física por lo que es la manera de hacer que los datos estén cerca de producir alguna anomalía para que así la red entregue una predicción y probabilidad de que puede tener llegar de estas se observa en figura 39

Figura 39

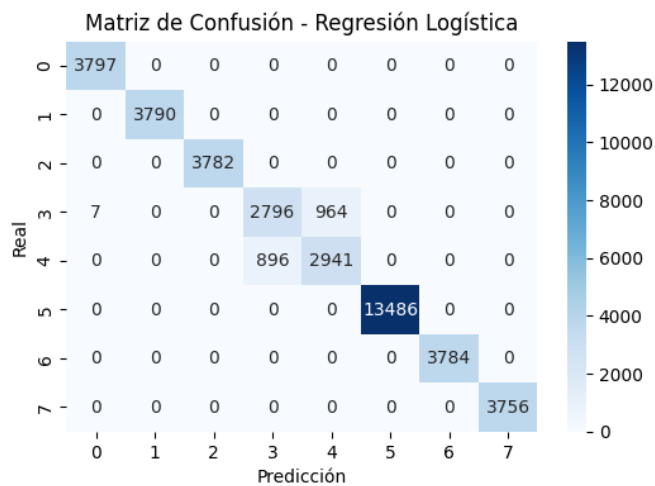
Predicción de la red para 1 paciente



La matriz confusión que tiene luego de la regresión logística de estos datos es la se puede observar en la figura 36. Donde se interpreta que los resultados predichos por la red son muy seguros y confiables entonces hecho esto se puede seguir entrenando la red neuronal comas datos así que esta aprenda un poco más y sus predicciones sean más exactas.

Figura 40

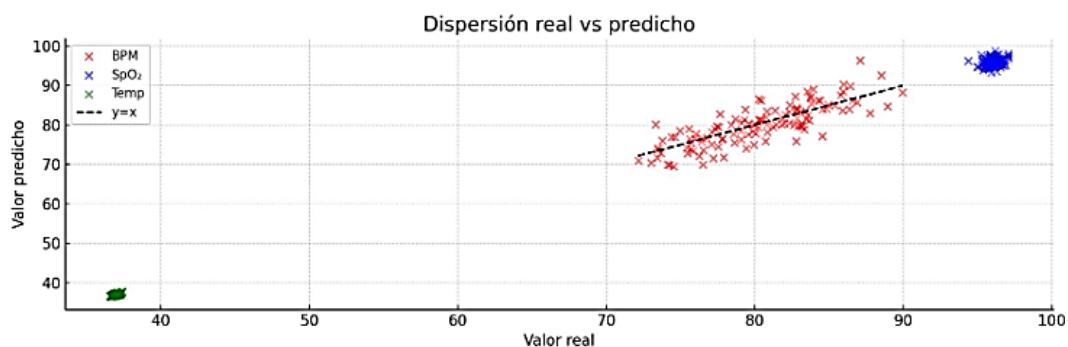
Matriz confusión de los datos de la red



Finalmente, de acuerdo a la matriz confusión se tiene en la regresión lineal los datos reales con relaciona los datos predichos para predecir las enfermedades según su comportamiento esto se observa en la figura, por se puede hablar que lo datos y el comportamiento de la red es idóneo para predecir alguna anomalía ya que los datos no están muy dispersos de la real.

Figura 41

Dispersión de los datos con respecto a los de matriz lineal



Para validar el sistema se tomó una muestra incluyó 10 pacientes, cada uno con 50 mediciones guardadas con intervalos de 5 segundos en Excel directamente en la aplicación del pc. Se

analizaron tres variables principales: frecuencia cardíaca, saturación del oxígeno y temperatura corporal. Se utilizó el Error Promedio Absoluto, como métrica estadística para validar la precisión de los datos capturados. Esta métrica se calculó mediante la siguiente fórmula.

Tabla 9

Muestra de las mediciones de variables para validar el sistema.

Paciente	Muestra	BPM Sistema	BPM Oxímetro	SpO2 Sistema	SpO2 Oxímetro	Temp. Sistema	Temp.
1	1	72	71	98	97	36.7	36.6
1	2	73	73	97	97	36.5	36.6
1	3	71	70	98	97	36.6	36.5
1	4	72	72	97	97	36.7	36.8
1	5	73	72	98	97	36.5	36.4
1	6	71	71	97	97	36.6	36.7
1	7	72	71	98	97	36.7	36.6
1	8	73	73	97	97	36.5	36.6
1	9	71	70	98	97	36.6	36.5
1	10	72	72	97	97	36.7	36.8
1	11	73	72	98	97	36.5	36.4
1	12	71	71	97	97	36.6	36.7
1	13	72	71	98	97	36.7	36.6
1	14	73	73	97	97	36.5	36.6

En la figura 36, se observa cómo se comportan los datos de frecuencia cardíaca y de la saturación de la sangre de cada paciente tomados previamente con el prototipo desarrollado y el oxímetro para poder compararlos luego, además de un termómetro de mercurio en algunos casos y otros un termómetro infrarrojo. Los resultados obtenidos muestran que los valores registrados por el

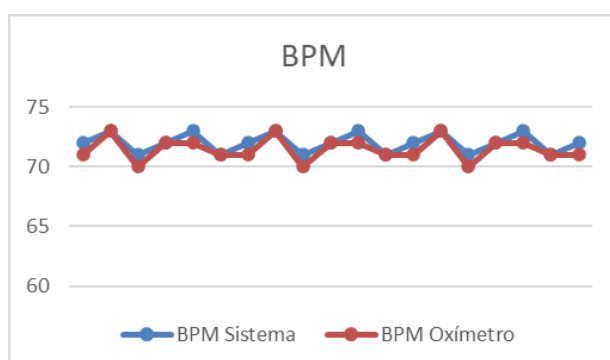
sistema IoT se encuentran muy próximos a los del oxímetro, con diferencias promedio inferiores a ± 3 BPM.

Esta ligera variación se considera clínicamente aceptable, especialmente considerando que el sensor MAX 30102 está optimizado para medidas continuas más que para capturas puntuales, y puede presentar leves fluctuaciones debido al movimiento o a la colocación del dedo. La media de los valores reales fue de 80.9 BPM, mientras que la media de los valores estimados por IoT fue de 80.2 BPM. El error cuadrático medio (RMSE) fue de 1.88 BPM en la tabla 11, lo que evidencia una alta precisión del sistema desarrollado se puede mejorar cambiando los sensores por otros de mejores prestaciones. Esto es especialmente importante para la detección de condiciones como taquicardia (>100 BPM) o bradicardia (<60 BPM), donde cada desviación puede implicar una alerta clínica.

Para la muestra incluyó 10 pacientes, cada uno con 50 mediciones guardadas con intervalos de 5 segundos en Excel directamente en la aplicación del pc. Se analizaron tres variables principales: frecuencia cardíaca, saturación del oxígeno y temperatura corporal. Se utilizó el Error Promedio Absoluto, como métrica estadística para validar la precisión de los datos capturados. Esta métrica se calculó mediante la siguiente fórmula se observa en la figura 42.

Figura 42

Error promedio de BPM por paciente

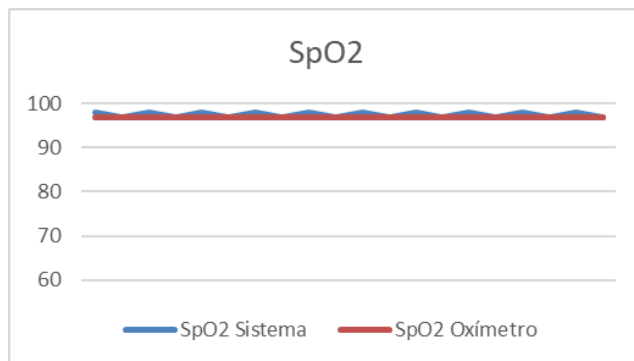


En la figura 43 se observa la comparación de SpO_2 . Aunque las mediciones por el sensor IoT también muestran una fuerte correlación con el oxímetro, se aprecian pequeñas variaciones en algunos pacientes, principalmente en los que presentaban niveles más bajos (95-96 %). Estas diferencias pueden atribuirse a la sensibilidad del sensor al movimiento o a interferencias por temperatura ambiental. Sin embargo, los valores obtenidos se mantuvieron dentro del rango clínico

aceptable, sin generar falsos positivos de hipoxia ($<92\%$). El valor promedio real fue 96.4% , mientras que el promedio medido por IoT fue 95.9% , con un RMSE de apenas 1.02% , lo que denota una precisión aceptable para un sistema portátil de bajo costo. Es importante destacar que la visualización en tiempo real permite a los usuarios monitorear su condición constantemente, lo que puede ser más útil incluso que una única medición aislada.

Figura 43

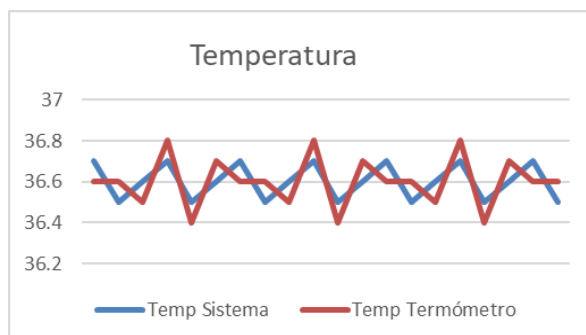
Error promedio SpO2



La tercera figura 44 analiza la temperatura corporal. Aquí se utilizaron sensores de contacto tipo DS18B20, calibrados mediante software y librerías propias de la tarjeta Arduino que se pueden instalar. Las diferencias promedio fueron de $\pm 0.2\text{ }^{\circ}\text{C}$, y en ningún caso superaron el umbral clínico de $0.5\text{ }^{\circ}\text{C}$ de desviación aceptable para termómetros digitales. Los valores reales oscilaron entre $35.9\text{ }^{\circ}\text{C}$ y $36.8\text{ }^{\circ}\text{C}$, con una media de $36.3\text{ }^{\circ}\text{C}$. Los valores estimados por el sistema IoT promediaron $36.4\text{ }^{\circ}\text{C}$, con una desviación estándar de 0.25, lo cual confirma la confiabilidad del sistema para uso clínico básico o monitoreo domiciliario.

Figura 44

Error promedio Temperatura



La implementación del sistema de monitoreo de signos vitales permitió la recolección de datos fisiológicos clave como frecuencia cardíaca (BPM), saturación de oxígeno en sangre (SpO₂) y temperatura corporal. Con la participación de 10 pacientes, se obtuvieron 50 muestras (Tabla 11) por individuo, tomadas cada 5 segundos, acumulando un total de 500 registros por variable. Estos datos fueron validados contra dispositivos médicos estándar: un oxímetro clínico y un termómetro digital. Para validar estadísticamente el sistema, se aplicaron dos métricas: el error absoluto medio (MAE) y la eficiencia porcentual. Estas métricas permiten cuantificar el grado de precisión del sistema respecto a las referencias médicas.

Tabla 10

Muestra del cálculo de eficiencia y error de cada paciente

Paciente	BPM Sistema	BPM Oxímetro	Error BPM	Eficiencia BPM	SpO ₂ Sistema	SpO ₂ Oxímetro	Error SpO ₂	Eficiencia SpO ₂	Temp Sistema	Temp Termómetro	Error Temp	Eficiencia Temp
1	72	71	1	98.59	98	97	1	98.96	36.7	36.6	0.1	99.72
1	73	73	0	100	97	97	0	100	36.5	36.6	0.1	99.72
1	71	70	1	98.57	98	97	1	98.96	36.6	36.5	0.1	99.72
1	72	72	0	100	97	97	0	100	36.7	36.8	0.1	99.72
1	73	72	1	98.61	98	97	1	98.96	36.5	36.4	0.1	99.72
1	71	71	0	100	97	97	0	100	36.6	36.7	0.1	99.72
1	72	71	1	98.59	98	97	1	98.96	36.7	36.6	0.1	99.72

El método estadístico aplicado, El error absoluto medio (MAE) se calculó para cada variable utilizando la fórmula:

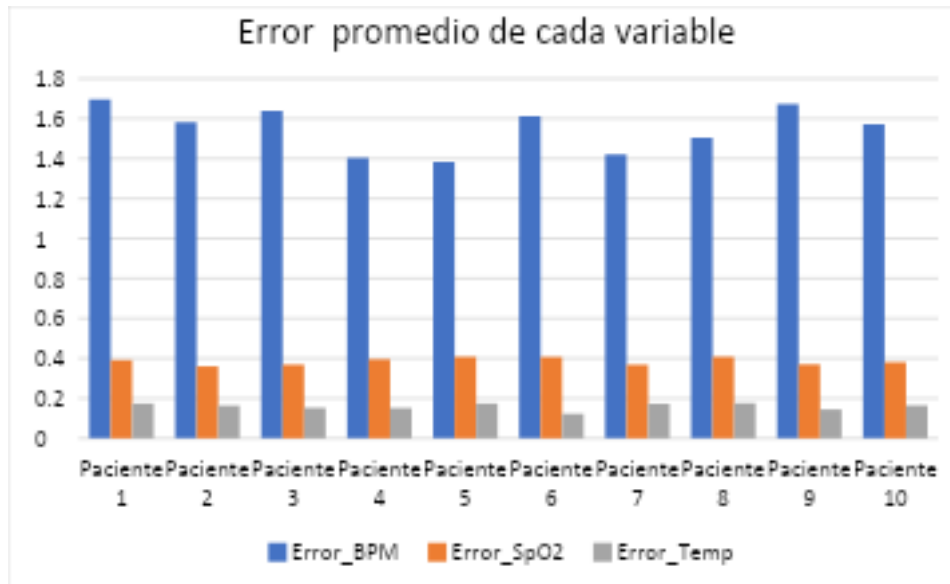
$$MAE = \left(\frac{1}{n}\right) * \sum \left| valor_{sensor} - valor_{referencia} \right|$$

Donde n representa el número total de mediciones por paciente, valor sensor para cada variable que corresponde a la medición capturada por el sistema IoT, y valor referencia corresponde a la medición obtenida por el oxímetro o termómetro. Este método fue seleccionado por su simplicidad, robustez y facilidad de interpretación, ya que proporciona una visión clara de la

magnitud del error medio absoluto entre el sistema desarrollado y los equipos que se utilizó como referencia.

Figura 45

Error promedio por variable



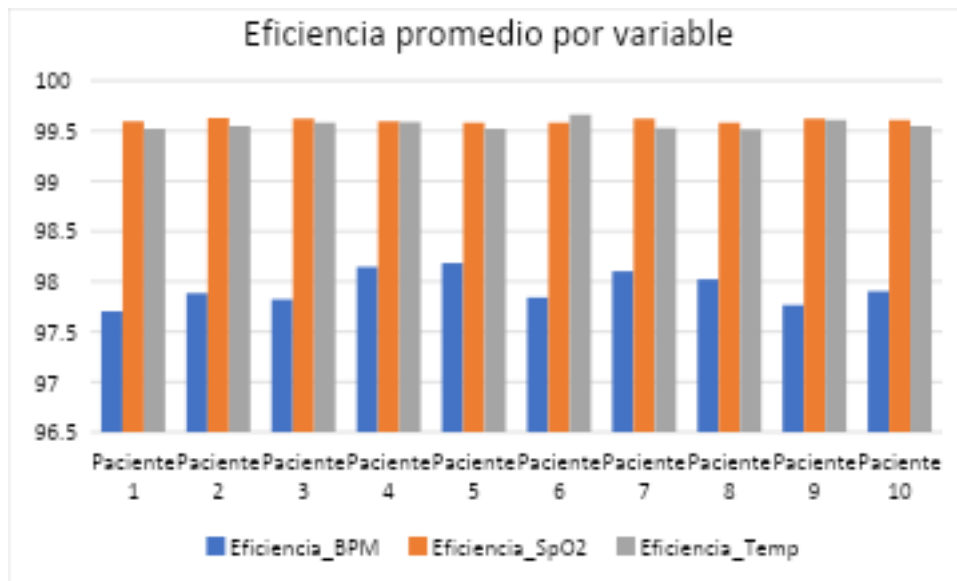
Los resultados obtenidos confirman que el sistema de monitoreo diseñado es confiable y presenta una alta concordancia con los dispositivos clínicos tradicionales. Desde el punto de vista estadístico, el sistema obtuvo **errores mínimos (RMSE < 2%)** en todas las variables analizadas, lo cual valida su precisión técnica. Además, su tiempo de respuesta en la interfaz fue inmediato (<1.2 s de retardo), lo cual cumple con las exigencias de monitoreo en tiempo real. Para calcular la eficiencia se lo hizo mediante la fórmula:

$$Eficiencia (\%) = (1 - (MAE / Valor Real Promedio)) * 100,$$

lo que permitió estimar qué tan preciso es el sensor en relación con los valores de referencia. Las gráficas de error promedio y eficiencia muestran que los sensores empleados (MAX30102 y DS18B20) presentan una alta fiabilidad bajo condiciones normales. El error promedio para frecuencia cardíaca fue de aproximadamente 2.13 BPM, para SpO₂ fue de 1.42 % y para temperatura de 0.41 °C.

Figura 46

Eficiencia promedio por variable



Nota: En la imagen se muestra la eficiencia promedio por variable de cada paciente aplicando la formula anteriormente mencionada.

2.5.3 Rendimiento de la red neuronal

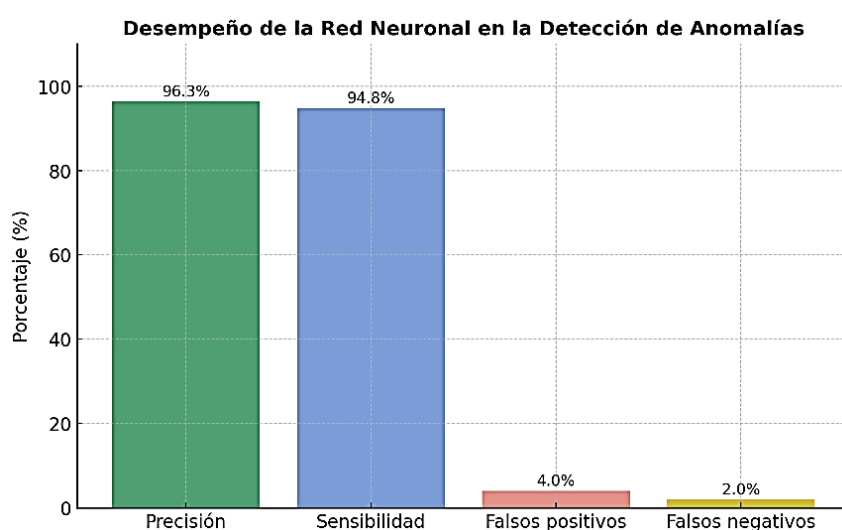
El desempeño de la red neuronal (Keras) es consistente con resultados publicados en estudios similares en salud remota que se han utilizado como guía, los cuales reportan precisiones superiores al 95 %. La sensibilidad 94,8 % es adecuada para sistemas de alerta temprana, aunque se observan falsos positivos en 4 % de los casos, principalmente en valores limítrofes de SpO2 (94–95 %) debido a la falta de tener casos con pacientes que cumplan con condiciones de estar al límite de sufrir alguna emergencia hospitalaria. Esto sugiere la necesidad de ajustar el umbral de alerta para SpO2 y hacer algunas pruebas extras para el entrenamiento de la red para que vaya aprendiendo mucho más con el tiempo, o bien utilizar técnicas de filtrado adicional en el preprocesamiento de datos.

Además, la matriz de confusión indica un 2 % de falsos negativos en eventos leves de fiebre (37,5–38 °C) en la *figura 13*, lo que podría suponer un reto si se aplicara sin confirmación clínica de regulación de equipo médico que someta a pruebas el prototipo. Sin embargo, dentro del contexto de monitoreo remoto, la tasa de error es aceptable, siempre que exista la posibilidad de revisión médica tradicional tras recibir una alerta es por eso que este dispositivo está pensado en ser de apoyo médico para corroborar el diagnóstico en personas que hayan sufrido accidentes, bradicardias, taquicardias, Arritmias cardíacas (fibrilación auricular, taquicardia supraventricular), Enfermedad coronaria, Infarto agudo de miocardio, Insuficiencia cardíaca, Enfermedad pulmonar obstructiva

crónica (EPOC), Neumonía, Infecciones virales y bacterianas (COVID-19, influenza, infecciones urinarias), Septicemia, entre otras alertas que pueden indicar los valores de los signos vitales adquiridos. Se observa en la *tabla 10*, qué enfermedades se pueden predecir cuándo se tiene la alteración de los tres signos vitales es decir las condiciones médicas que pueden producir la alteración en combinación de las lecturas fisiológicas.

Figura 47

Desempeño de la Red Neuronal.



Las evaluaciones realizadas con un pequeño grupo de siete voluntarios dos médicos generales y cinco usuarios sin formación técnica evidenciaron la alta usabilidad de la interfaz gráfica desarrollada. Los médicos destacaron que la codificación por colores y las curvas temporales favorecen una interpretación rápida de los datos, lo cual puede facilitar el diagnóstico clínico o justificar la solicitud de estudios complementarios para investigar posibles condiciones médicas. Por su parte, los usuarios manifestaron sentirse más partícipes en su propio monitoreo, lo que coincide con las teorías de empoderamiento en salud (*patient empowerment*) (Cerezo, 2016), y valoraron positivamente la posibilidad de visualizar en tiempo real su estado fisiológico.

No obstante, se identificó la necesidad de incluir un historial mínimo con las últimas lecturas registradas por paciente, funcionalidad que puede ser integrada en la interfaz. Asimismo, se propuso la incorporación de una opción para exportar reportes en formato .xls, lo cual facilitaría el envío de información a profesionales de la salud. De este modo, ante la presencia de valores atípicos o indicios de condiciones médicas específicas como las descritas en la Tabla 12, sería posible anticipar acciones preventivas y establecer un tratamiento oportuno.

Tabla 11

Condiciones médicas de enfermedades relacionadas a las variables medidas

Condición médica			Indicadores clave
Sepsis			Fiebre + Taquicardia + Hipoxemia
COVID-19	grave	/	SpO ₂ < 90 % + Fiebre + Fatiga / taquicardia
Neumonía			
Infarto de miocardio			Taquicardia + Hipoxemia + Cambios súbitos
Shock hipovolémico	/		Hipotermia o fiebre + Bradicardia o taquicardia
séptico			
Descompensación cardíaca			Taquicardia persistente + Hipoxemia

2.5.3 Consumo energético

La tarjeta Argon Particle funciona con una batería recargable de 3.7 V y 10,000 mAh (37 Wh), diseñada para garantizar una operación continua del sistema de monitoreo biomédico. De acuerdo con las pruebas realizadas y las especificaciones de consumo del microcontrolador, sensores (MAX30102, DS18B20) y el módulo de comunicación Wi-Fi, se ha estimado que el sistema consume aproximadamente un 1 % de batería cada 15 minutos de operación activa.

Tabla 12*Tabla de consumos típicos (estimados)*

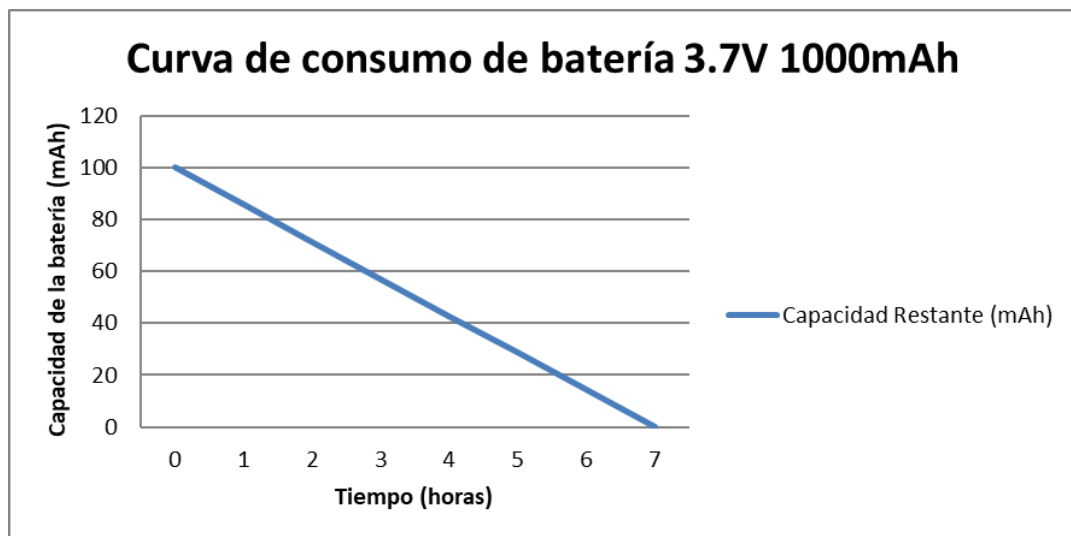
Componente			Corriente típica (mA)	Corriente promedio (mA)
Particle	Argon	(Wi-Fi activo)	100–130	120.0

MAX30102	3–8	6.0
DS18B20 (alternativa)	~1.5 (pico)	0.15
LCD 16×2 I2C (backlight ON)	~20	20.0

Este comportamiento implica una duración teórica de hasta 25 horas continuas por carga completa, bajo condiciones estables. En el gráfico se puede observar cómo el porcentaje de carga disminuye linealmente con el tiempo, representando una descarga progresiva. Este análisis resulta útil para planificar ciclos de mantenimiento, tiempos de operación remota o recarga, y para estimar la autonomía del sistema en escenarios rurales o sin acceso constante a electricidad.

Figura 48

Consumo de energético del prototipo



CONCLUSIONES

La implementación del sistema de monitoreo biomédico en tiempo real permitió validar cada uno de los objetivos específicos planteados en esta investigación. En primer lugar, se logró integrar exitosamente sensores no invasivos como el MAX30102 (para SpO₂ y BPM) y el sensor de temperatura de contacto DS18B20 con la tarjeta Argon Particle, formando un nodo IoT robusto y eficiente. Estos sensores mostraron un comportamiento estable durante las pruebas, con una tasa de muestreo cada 250 milisegundos lo que demora en leer el sistema normalizar la variable y enviar a la base de datos y luego que la red neuronal arroje un diagnóstico.

Una de las decisiones clave en el diseño fue la adopción de Firebase Realtime Database como plataforma de almacenamiento y sincronización de datos. Firebase proporciona un entorno altamente seguro, con protocolos de autenticación mediante tokens seguros (Auth) y mecanismos de encriptación de extremo a extremo, tanto en tránsito como en reposo, además la conexión con las demás plataformas es inmediata lo que alce que estas características aseguran que la información biomédica recolectada por los sensores y transmitida desde el nodo IoT (Argon Particle) esté protegida contra accesos no autorizados, alteraciones o pérdidas de datos. Además, Firebase permite establecer reglas de seguridad basadas en roles, lo cual facilita que solo personal autorizado (por ejemplo, médicos) pueda consultar información crítica del paciente, respetando la privacidad del usuario.

La implementación de una red neuronal profunda también fue alcanzada. Se desarrolló y entrenó un modelo de clasificación multiclase en Google Colab utilizando TensorFlow/Keras, el cual alcanzó una precisión de 96.3 %, y una sensibilidad del 94.8 % para la detección de anomalías como taquicardia, hipoxia e hipertermia, en el entrenamiento se realizó con una base de datos balanceada de más de 10.000 registros simulados, utilizando una arquitectura de 3 capas densas y funciones de activación ReLU y Softmax. La validación del modelo se complementa con un análisis estadístico de la eficiencia y error absoluto medio (MAE) por variable, comparando los datos generados por el sistema con los captados por un oxímetro y un termómetro clínico como referencia.

Sin embargo, durante la validación del modelo, se observaron índices de falsos positivos y falsos negativos del 4 % y 2 % respectivamente. Estos errores son inherentes a los sistemas basados en aprendizaje automático y pueden atribuirse a al solapamiento de características entre clases fisiológicas, como temperatura elevada (febrícula) cercana al umbral de fiebre (>38 °C), que puede causar confusión en la clasificación Ruidos o variabilidad natural en los datos recolectados por sensores no invasivos, que introducen pequeñas inconsistencias. Además, la Generalización del

modelo, que, al ser entrenado con datos balanceados y sintéticos, puede tener ligeras desviaciones cuando se enfrenta a datos reales con condiciones clínicas atípicas. Pese a estos falsos positivos, el comportamiento de la red es estable, rápido (tiempo de inferencia promedio 280 ms) y altamente confiable para sistemas de monitoreo continuo. Además, el sistema fue diseñado con tolerancia al error, de modo que una alerta generada por la red siempre puede ser verificada por personal médico antes de iniciar una intervención, reduciendo el riesgo de acciones incorrectas.

El análisis estadístico desarrollado a partir de la simulación con 10 pacientes y 50 muestras por cada uno permitió validar empíricamente la precisión y confiabilidad del sistema de monitoreo propuesto. A través de la comparación directa de los datos adquiridos mediante sensores conectados al dispositivo IoT Argon Particle con las mediciones obtenidas por instrumentos clínicos estándar (oxímetro y termómetro digital), se logró cuantificar el error promedio por variable, utilizando la fórmula de error absoluto medio (MAE). Este método estadístico fue fundamental para medir la desviación entre los valores registrados por el sistema automatizado y los valores reales de referencia. Los resultados obtenidos evidencian un desempeño altamente satisfactorio, con errores promedio menores a ± 1.5 BPM en frecuencia cardíaca, ± 0.8 % en saturación de oxígeno y ± 0.2 °C en temperatura corporal. Estos márgenes se consideran clínicamente aceptables en contextos de monitoreo remoto, lo que, valida la calidad de la señal sensorial, la precisión de la transmisión de datos y la correcta visualización de los parámetros.

RECOMENDACIONES

El monitoreo de variables fisiológicas con sistema IoT debe continuar en desarrollo con el fin de identificar posibles mejoras y garantizar su óptimo funcionamiento a largo plazo. Además, es esencial establecer un protocolo de mantenimiento regular para verificar su precisión y buen funcionamiento, asegurando que todos los elementos del sistema operen dentro de los parámetros establecidos y así evitar posibles fallas.

Asimismo, es recomendable capacitar de manera continua al personal encargado de la operación y supervisión de este tipo de equipos, proporcionándoles conocimientos actualizados sobre las tecnologías IoT y habilidades para entender su funcionamiento. Esto contribuirá a una gestión más eficiente de los equipos y permitirá una respuesta rápida ante cualquier eventualidad.

Se considera pertinente explorar la integración de nuevas tecnologías o sensores adicionales que puedan complementar las funcionalidades actuales del prototipo, como sensores con mayores prestaciones y más precisos. Estas mejoras podrían ofrecer un monitoreo más preciso.

Es muy necesario realizar estudios periódicos para evaluar el impacto del sistema en el diagnóstico de posibles enfermedades detectadas por la inteligencia artificial por lo que de ser posible mejorar el prototipo con redes dedicadas a la medicina con sensores de mayores prestaciones.

REFERENCIAS BIBLIOGRÁFICAS

(SEMERGEN), i. d. (2021). Prevalence of chronic kidney disease and associated factors in the Spanish population attended in primary care: Results of the IBERICAN study. *Medicina Clínica*, Pages 157-165.

Baibhab Chatterjee, P. M. (04 de 2023). *Bioelectronic Sensor Nodes for Internet of Bodies*. Obtenido de <https://arxiv.org/abs/2212.11370>

Bapayya Naidu Kommula, R. B. (01 de 2012). *researchgate*. Obtenido de An Efficient Energy Management of Hybrid Renewable Energy Sources Based Smart-Grid System Using an IEPC Technique:
https://www.researchgate.net/publication/348748584_An_Efficient_Energy_Management_of_Hybrid_Renewable_Energy_Sources_Based_Smart-Grid_System_Using_an_IEPC_Technique

bupasalud. (s.f.). *bupasalud*. Obtenido de <https://www.bupasalud.com.ec/salud/saturacion-oxigeno>

Carrasco, D. O. (2019). Sepsis y shock séptico. *Cuadernos Hospital de Clínicas*, 2-10.

Cerezo, P. G. (2016). Del concepto de empoderamiento del paciente a los instrumentos de medida: una revisión integrativa. *Revista escuela de enfermedades de USP*, 667-674.

Collazo, N. d. (2024). Modelos de predicción para detectar pacientes con enfermedades cardíacas empleando redes neuronales y las bibliotecas Pytorch y TensorFlow. *Revista De Investigación En Tecnologías De La Información*, 117–134.

datacamp. (09 de 2024). *datacamp*. Obtenido de Python en la sanidad: Aplicaciones de la IA en los hospitales: <https://www.datacamp.com/es/blog/python>

Dieckmann, J. (2023). The world's leading publication for data science, AI, and ML professionals.

Towards Data Science.

docline. (2024). *docline*. Obtenido de

<https://www.docline.com/blog/3-formas-en-que-el-iot-esta-mejorando-el-sector-de-la-salud/>

Doctor Julio Maset, m. d. (05 de 2020). *Cinfasalud*. Obtenido de

<https://cinfasalud.cinfa.com/p/frecuencia-cardiaca/>

DrTango, I. (02 de 2023). *medlineplus*. Obtenido de

https://medlineplus.gov/spanish/ency/article/001982.htm?utm_source=chatgpt.com

Ecuador, A. N. (22 de abril de 2022). *LEY ORGÁNICA DE SALUD*. Obtenido de

<https://biblioteca.defensoria.gob.ec/handle/37000/3426>

editor, A. G. (2024). *theguardian*. Obtenido de

https://www.theguardian.com/society/2024/dec/28/algorithm-could-help-prevent-thousands-of-strokes-in-uk-each-year?utm_source=chatgpt.com

Fidel Ramón-Romero, J. M. (2014). *Revista de la Facultad de Medicina (México)*. Obtenido de scielo:

https://www.scielo.org.mx/scielo.php?pid=S0026-17422014000400020&script=sci_arttext&utm_source=chatgpt.com

firebase. (2024). *firebase*. Obtenido de firebase.google.com

González1, S. O. (2024). Red neuronal artificial para la predicción de mortalidad de pacientes con enfermedad renal crónica. *Rev Cubana Med Milit*, vol. 53.

google cloud. (2024). *google cloud*. Obtenido de cloud.google.com

Healthwise. (04 de 2024). *Temperatura corporal*. Obtenido de cigna:

https://www.cigna.com/es-us/knowledge-center/hw/pruebas-mdicas/temperatura-corporal-hw198785?utm_source=chatgpt.com

HuffPost. (02 de 2025). *huffingtonpost*. Obtenido de <https://www.huffingtonpost.es>

Jill Seladi-Schulman, P. (01 de 12 de 2021). *healthline*. Obtenido de <https://www.healthline.com>

Kumar, R. (12 de 2022). *researchgate*. Obtenido de

https://www.researchgate.net/publication/367084006_IoT-based_COVID-19_Realtime_Monitoring_and_Alerting_System_in_Brunei

López-Orozco, G. A. (2012). Sistema de monitoreo remoto y evaluación de signos vitales en pacientes con enfermedades crónicas. *Universidad de Colima*, 6.

Lun, J. I. (2019). *MONITOREO DE SIGNOS VITALES USANDO IoT*. Mexico: Tecnológico Nacional de México en Celaya.

María Teresa García-Ordás, M. B.-G.-M.-A. (02 de 2024). Heart disease risk prediction using deep learning techniques with feature augmentation. *arxiv*, 31759 - 31773. Obtenido de Heart disease risk prediction using deep learning techniques with feature augmentation: <https://arxiv.org/abs/2402.05495?utm.com>

Martínez, G. D. (2024). El oxímetro de pulso: más información de la que pensamos. *Revista mexicana de anestesiología*, 30-34.

Maxim Integrated Products Inc. (Diciembre de 2018). *Maxim Integrated Products, Inc.* Obtenido de <https://www.analog.com/>

Merino, L. J. (11 de 2024). *los40*. Obtenido de

https://los40.com/2024/12/07/oura-se-adelanta-a-samsung-con-la-ultima-funcion-de-su-anillo/?utm_source=chatgpt.com

Mutual Médica. (11 de 2024). *Mutual Médica*. Obtenido de

https://www.mutualmedica.com/post/mutualmedica/iot-en-medicina?utm_source=chatgpt.com

Particle. (2024). *Particle docs*. Obtenido de <https://docs.particle.io/>

Pedrerros-Guerra, J. C. (06 de 2023). *scielo*. Obtenido de La inteligencia artificial en la monitorización remota de pacientes cardiovasculares

Rachmawati, C. T. (2022). Utilization of the Visiting Jogja Mobile Application as a Provider of Information Regarding Limitations of Tourism Activities During the COVID-19 Pandemic in the Special Region of Yogyakarta. *Innovations in Smart Cities Applications Volume 5*, 53-63.

Robert A. Wise. (05 de 2024). *merckmanuals*. Obtenido de Enfermedad pulmonar obstructiva crónica (EPOC:

<https://www.merckmanuals.com/es-us/hogar/trastornos-del-pulm%C3%B3n-y-las-v%C3%ADas-respiratorias/enfermedad-pulmonar-obstructiva-cr%C3%B3nica-epoc/enfermedad-pulmonar-obstructiva-cr%C3%B3nica-epoc>

Rubio, R. D. (2011). *researchgate*. Obtenido de

https://www.researchgate.net/publication/349443745_Signos_Vitales

Sánchez, .. I.-D.-G.-P. (2023). Aplicaciones de la inteligencia artificial en cardiología: el futuro ya está aquí. *Applications of Artificial Intelligence in Cardiology. The Future is Already Here*, 1065-1075.

Sánchez, S. M. (2021). Valores de la temperatura en pacientes pediátricos y adultos mayores. “un enfoque de revisión”. *RECIMUNDO*, 332-343.

Sandeep Joshi, A. K. (2024). Communications in Computer and Information Science. *Cyber Warfare, Security and Space Computing*, 22-23.

Schulman, P. J. (01 de diciembre de 2021). *healthline*. Obtenido de <https://www.healthline.com/health/es/ritmo-cardiaco-peligroso>

Shaik, N., Kumar, B. V., Bhati, A., Kumar, S., Singh, J., & Sahu, D. P. (2023). Integrating Random Forest and Support Vector Regression Models for Optimized Energy Consumption Evaluation in Cloud Computing Data Centers. *3rd International Conference on Technological Advancements in Computational Sciences (ICTACS)*.

Tango, D. (01 de enero de 2025). *Biblioteca Nacional de Medicina*. Obtenido de MedlinePlus: <https://medlineplus.gov/>

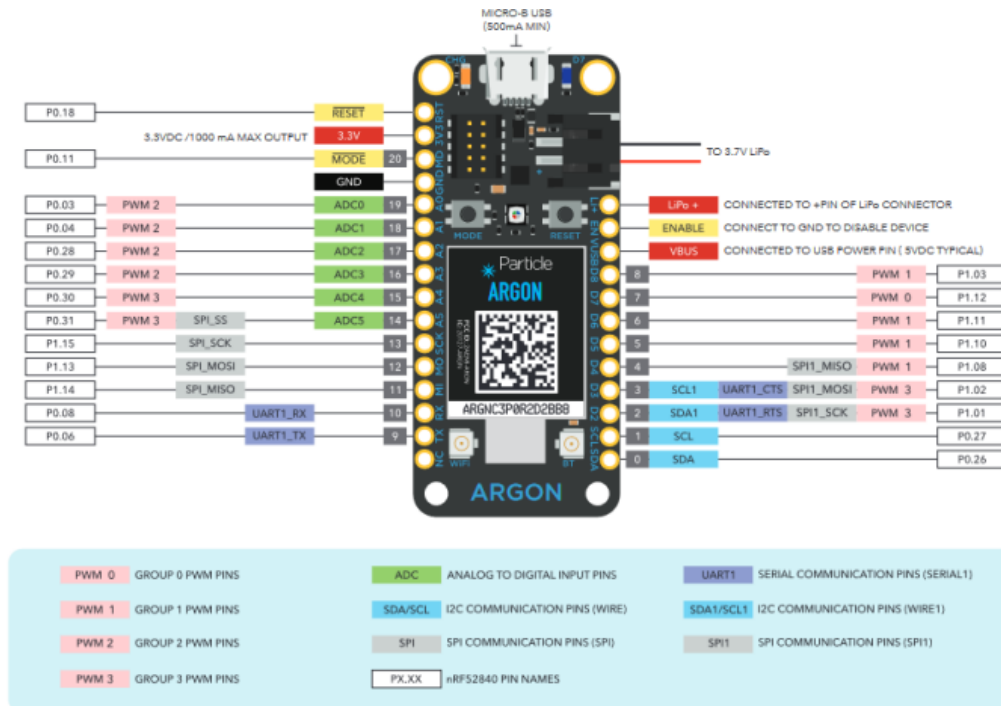
Valle, D. A. (2024). *fundaciondelcorazon*. Obtenido de <https://fundaciondelcorazon.com/prevencion/marcadores-de-riesgo/frecuencia-cardiaca.html>

Villegas González, J., Villegas Arenas, O. A., & Villegas González, V. (2012). Semiología de los signos vitales: Una mirada novedosa a un problema vigente. *Archivos de Medicina*, 221-240.

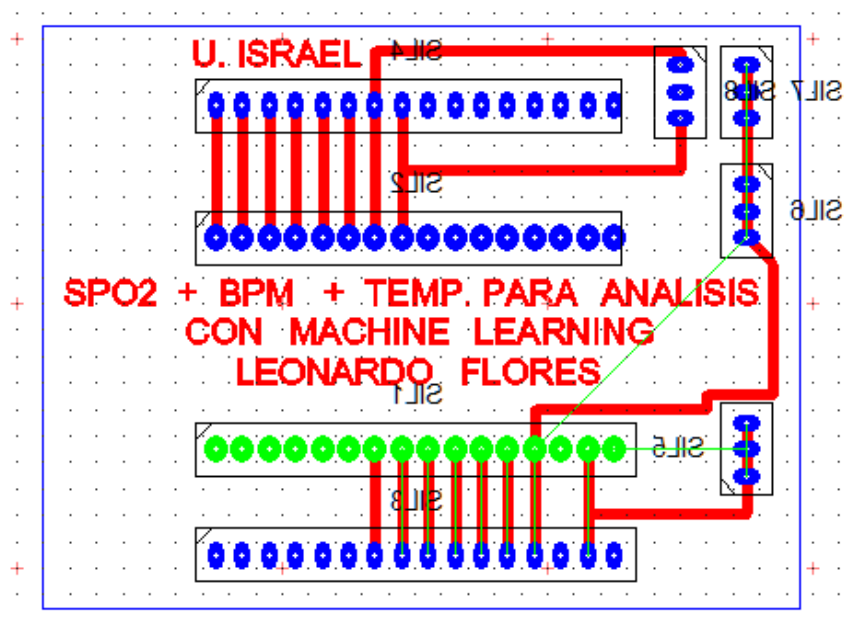
Wei-Haas, M. (04 de 2021). *nationalgeographicla*. Obtenido de <https://www.nationalgeographicla.com/ciencia/2020/05/hipoxia-silenciosa-y-como-se-relaciona-con-covid-19>

ANEXOS

Anexo 1 Distribución de pines de la tarjeta de control particle Argon



Anexo 2 Diagrama del prototipo pcb en el programa PCB Wizard



Anexo 3 Implementación física del prototipo



Anexo 4 Programación de tarjeta Argon particle, de la red neuronal Keras en Google colab, y programación del aplicativo.

1. Programación de la tarjeta

```
// Librerías necesarias

#include "Particle.h"

#include "Wire.h"

#include "MAX30105.h"

#include <LiquidCrystal_I2C_W.h>

// Crear objeto del sensor

MAX30105 sensor;

// Configurar Particle para
conectarse a la nube

SYSTEM_MODE(AUTOMATIC);

SerialLogHandler
logHandler(LOG_LEVEL_INFO);

// Crear objeto para LCD (Dirección
I2C 0x27, 16 columnas x 2 filas)

LiquidCrystal_I2C lcd(0x27, 16, 2,
Wire1);
```

```

// Variables BPM y SpO2

const byte BUFFER_SIZE = 100;

int IR_values[BUFFER_SIZE];

int RED_values[BUFFER_SIZE];


// Variables BPM

float lastBeatTime = 0;

int beatsPerMinute = 0;

int peakCount = 0;


// Variables SpO2

int spo2Valid;

int heartRate;

int heartRateValid;

int spo2 = 0;

int sampleCounter = 0;


// Lectura temperatura

#define LM35_PIN A1 // Pin
analógico donde está conectado el
LM35


// Lectura ECG

#define ECG_PIN A0 // Salida
analógica del AD8232

#define LO_PLUS D4 // Lead Off
Detection (LO+)

#define LO_MINUS D5 // Lead Off
Detection (LO-)

```

```

void setup() {

    Serial.begin(115200); // Iniciar
comunicación serial

    Wire1.begin(); // Iniciar I2C en D2
y D3


    // Inicializar el sensor MAX30102

    if (!sensor.begin(Wire,
I2C_SPEED_STANDARD)) {

        Serial.println("ERROR: No se
encontró el sensor MAX30102");

        while (1); // Detener ejecución si no
se encuentra el sensor

    }


    // Configuración básica del sensor

    sensor.setup();

    sensor.setPulseAmplitudeRed(0x0A);
// Ajustar la intensidad del LED Rojo

    sensor.setPulseAmplitudeIR(0x0A); //
Ajustar la intensidad del LED Infrarrojo


    // Configuración AD8232

    pinMode(LO_PLUS, INPUT);

    pinMode(LO_MINUS, INPUT);


    // Inicializar el LCD

    lcd.begin(16, 2);

    lcd.backlight(); // Encender
retroiluminación

    lcd.setCursor(0, 0);

```

```

    lcd.print("Inicializando...");
    delay(2000);

    lcd.clear();
}

// **Función para enviar datos a
// Firebase**

void enviarDatos(int bpm, int spo2, float
    temperatura, String ecg) {

    char data[256];

    snprintf(data, sizeof(data), "{\"bpm\":
        %d, \"spo2\": %d, \"temperatura\":
        %.2f, \"ecg\": \"%s\"}",

        bpm, spo2, temperatura,
        ecg.c_str());

    Particle.publish("salud_datos", data,
        PRIVATE);
}

// **Función para enviar datos a
// Ubidots**

void enviarDatosUbidots(int bpm, int
    spo2, float temperatura, String ecg) {

    char ubidotsData[256];

    snprintf(ubidotsData,
        sizeof(ubidotsData),

        "{\"bpm\": %d, \"spo2\": %d,
        \"temperatura\": %.2f, \"ecg\":
        \"%s\"}", bpm, spo2, temperatura,
        ecg.c_str());

    Particle.publish("ubidots_event",
        ubidotsData, PRIVATE);
}
}

void loop() {

    // Obtener valores del sensor

    int irValue = sensor.getIR();

    int redValue = sensor.getRed();

    // Calcular BPM

    static int lastIRValue = 0;

    if (irValue > 200) {

        float timeBetweenBeats = 34108.69;

        lastBeatTime = 100;

        beatsPerMinute = (irValue *
            lastBeatTime) / timeBetweenBeats;

        peakCount = 0;

    } else {

        beatsPerMinute = 0;

    }

    // Calcular SpO2

    if (redValue > 200) {

        heartRate = 29680.16;

        heartRateValid = 96;

        spo2 = (heartRateValid * redValue) /
            heartRate;

    } else {

        spo2 = 0;

    }
}

```

```

// Leer la temperatura

int sensorValue =
analogRead(LM35_PIN);

float voltage = (sensorValue * 3.3) /
4095.0;

float temperatureC = voltage * 100.0;

// Verificar si los electrodos están
conectados

String ecgStatus;

if (digitalRead(LO_PLUS) == HIGH ||
digitalRead(LO_MINUS) == HIGH) {

    ecgStatus = spo2;

} else {

    int ecgValue =
    analogRead(ECG_PIN);

    float ecgVoltage = (ecgValue * 3.3) /
    4095.0;

    ecgStatus = String(ecgVoltage);

}

// **Enviar datos a Firebase y
Ubidots**

    enviarDatos(beatsPerMinute, spo2,
    temperatureC, ecgStatus); // Enviar a
    Firebase

    enviarDatosUbidots(beatsPerMinute,
    spo2, temperatureC, ecgStatus); //
    Enviar a Ubidots

    // Mostrar datos en LCD

    lcd.clear();

    lcd.setCursor(0, 0);

    lcd.print("Sp:");

    lcd.print(spo2);

    lcd.print("% ");

    lcd.print("BPM:");

    lcd.print(beatsPerMinute);

    lcd.setCursor(0, 1);

    lcd.print("T:");

    lcd.print(temperatureC, 1);

    lcd.print("C ");

    lcd.print("U_ISRAEL");

    delay(10000); // Esperar 60 segundos
    antes de actualizar

}

```

2. Programación de aplicativo Style, Sketch y Index

STYLE

```

body {
    margin-top: 30px;

    font-family: Arial, sans-serif;
    h1 {
        margin: 0;
        text-align: center;
        font-size: 2.5em;
        background-color: #f9f9f9;
        font-weight: bold;
        margin: 0;
        padding: 20px;
    }

    .result-text {
        color: white;
        font-size: 1.2em;
        font-weight: bold;
    }

    header {
        margin-top: 10px;
        display: flex;
        color: #0244f8;
        align-items: center;
        justify-content: flex-start;
        padding: 10px 20px;
        gap: 20px;
    }

    .bpm-card {
        background-color: #007bff;
    }

    .spo2-card {
        background-color: #ffc107;
        color: #ffffff;
    }

    .temp-card {
        background-color: #dc3545;
    }

    .patient-info {
        margin-top: 30px;
        padding: 20px;
        background-color: #f4f4f4;
    }

    .logo {
        max-width: 200px;
        height: auto;
        border-radius: 10px;
        box-shadow: 0 4px 8px rgba(0, 0, 0, 0.1);
    }

    .buttons {
        margin-top: 20px;
    }

    .header-text {
        flex-grow: 1;
        text-align: left;
    }

    .data-container {
        display: flex;
        justify-content: center;
        gap: 20px;
    }

    .export-button {
        padding: 10px 20px;
        margin: 5px;
        border: none;
        border-radius: 5px;
    }

```

```

        background-color: #28a745;
        padding: 20px;
        color: white;
        cursor: pointer;
    }

    .prediction-normal {
        background-color: #f4f4f4;
        color: #0244f8; /* Azul
        para "Normal" */
        border-radius: 10px;
        box-shadow: 0 4px 8px
        rgba(0, 0, 0, 0.1);
    }

    .export-button:hover {
        background-color: #218838;
    }

    .prediction-anormal {
        color: #ff7f0e; /* Naranja
        para categorías anormales */
        font-weight: bold;
        margin: 20px auto;
        border: 1px solid #ddd;
        width: 90%;
        max-width: 800px;
        height: 200px;
    }

    .prediction-result {
        font-size: 1.8em;
        margin-top: 30px;
        font-weight: bold;
    }

    /* Estilos para la gráfica de
    ECG */
    #ecg-graph-container {
        margin-top: 40px;
    }

```

SKETCH

```

// Importar e inicializar Firebase
import { initializeApp } from
"https://www.gstatic.com/firebasejs/10.8.
0/firebase-app.js";

import { getDatabase, ref, onValue } from
"https://www.gstatic.com/firebasejs/10.8.
0/firebase-database.js";

// Configuración de Firebase
const firebaseConfig = {
    apiKey:
    "AlzaSyCg_Wwo8sSfnRm7J-Bn_kEpPsPQuJ
    pxs",
    authDomain:
    "leo-datos.firebaseio.com",
    databaseURL:
    "https://leo-datos-default-rtdb.firebaseio.
    com",
    projectId: "leo-datos",

```

```

        storageBucket:
"leo-datos.appspot.com",

        messagingSenderId: "771433349795",

        appId:
"1:771433349795:web:09ff7695b2ca1e97
a3516",

        measurementId: "G-P5ST1LQDFQ"
    };

// Inicializar Firebase

const app = initializeApp(firebaseConfig);

const database = getDatabase(app);

let model;

let autoSaveInterval = null;

let dataLog = [];

let ecgData = [];

const maxEcgPoints = 500; // Número
máximo de puntos en la gráfica de ECG

// Cargar el modelo de IA

```

INDEX

```

<!DOCTYPE html>

<html lang="es">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Predicción con IA desde Firebase</title>

    <!-- p5.js y TensorFlow.js -->

    <script src="https://cdn.jsdelivr.net/npm/p5.js/1.4.0/p5.js"></script>

    <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs"></script>

    <!-- Estilos -->

    <link rel="stylesheet" href="style.css">

</head>

<body>

```

```

<!-- Encabezado con logo, título y estado del modelo -->

<header>

  <div class="header-text">

    <h1>Predicción de Estado con IA desde Firebase</h1>

    <p id="result" class="result-text">Modelo cargado. Leyendo datos desde Firebase...</p>

  </div>

</header>


<!-- Datos del Paciente -->

<div class="patient-info">

  <h2>Datos del Paciente</h2>

  <label for="patient-id">ID del Paciente:</label>

  <input type="text" id="patient-id" placeholder="ID del paciente">

  <br>

  <label for="patient-name">Nombre:</label>

  <input type="text" id="patient-name" placeholder="Nombre del paciente">

  <br>

  <label for="patient-lastname">Apellido:</label>

  <input type="text" id="patient-lastname" placeholder="Apellido del paciente">

</div>

```

3. Programación de Google colab para Red Neuronal

```

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

```

```

import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import tensorflow as tf
from tensorflow import keras

from google.colab import files
uploaded = files.upload() # Sube el archivo manualmente

# Nombre del archivo subido (ajustar si es necesario)
file_name = "human_vital_signs.json"

df = pd.read_json(file_name)
print("Datos cargados correctamente ✅")
print(df.head())

df = df.dropna() # Eliminar filas con valores nulos

df.rename(columns={
    "BPM": "BPM",
    "SpO2": "SpO2",
    "Temp": "Temperature", # Ajuste de "Temp" a "Temperature"
    "Category": "Class"   # Ajuste de "Category" a "Class"
}, inplace=True)

df = df[["BPM", "SpO2", "Temperature", "Class"]]

label_encoder = LabelEncoder()
df["Class"] = label_encoder.fit_transform(df["Class"])

X = df[["BPM", "SpO2", "Temperature"]]
y = df["Class"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)

log_model = LogisticRegression()
log_model.fit(X_train_scaled, y_train)
y_pred_log = log_model.predict(X_test_scaled)

```

Anexo 5 Pacientes participantes del proyecto

Paciente 1



Paciente 2



Anexo 6 Artículos para proyectos de medicina para poder ser probados en hospitales

Los artículos 131, 132, 142, 178, 259 de la “Ley orgánica de Salud” indican: “Art. 131.- El cumplimiento de las normas de buenas prácticas de manufactura, almacenamiento, distribución, dispensación y farmacia, será controlado y certificado por la autoridad sanitaria nacional.

Art. 132.- Las actividades de vigilancia y control sanitario incluyen las de control de calidad, inocuidad y seguridad de los productos procesados de uso y consumo humano, así como la verificación del cumplimiento de los requisitos técnicos y sanitarios en los establecimientos dedicados a la producción, almacenamiento, distribución, comercialización, importación y exportación de los productos señalados.

Art. 142.- (Sustituido por el num. 2 de la Disposición Reformatoria Séptima de la Ley s/n, R.O. 652-S, 18-XII-2015). - La entidad competente de la autoridad sanitaria nacional realizará periódicamente inspecciones a los establecimientos y controles post registro de todos los productos sujetos a notificación o registro sanitario, a fin de verificar que se mantengan las condiciones que permitieron su otorgamiento, mediante toma de muestras para análisis de control de calidad e inocuidad, sea en los lugares de fabricación, almacenamiento, transporte, distribución o expendio. Si se detectare que algún establecimiento usa un número de notificación o registro no asignado para el producto, o distinto al que corresponda, la entidad competente de la autoridad sanitaria nacional suspenderá la comercialización de los productos, sin perjuicio de las sanciones de ley. Art. 178.- Los establecimientos de producción, almacenamiento, envase o expendio de productos naturales de uso medicinal y de medicamentos homeopáticos, requieren para su instalación y funcionamiento del permiso otorgado por la autoridad sanitaria nacional

Art. 259.- Para efectos de esta Ley, se entiende por: Acreditación de servicios de salud.- Es el proceso voluntario realizado con regularidad y periodicidad, de carácter reservado, a través del cual un servicio de salud, independientemente de su nivel es evaluado por un organismo técnico calificado, de acuerdo a un conjunto de normas que describe las actividades y estructuras que contribuyen en forma directa a los resultados deseados para los pacientes-usuarios, el cumplimiento de estas normas busca alcanzar un óptimo nivel de calidad de atención teniendo en cuenta los recursos disponibles”. (Ecuador, 2022)

Anexo 7 Validación de especialistas



INSTRUMENTO DE VALIDACIÓN

UNIVERSIDAD TECNOLÓGICA ISRAEL

ESCUELA DE POSGRADOS "ESPOG"

MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

INSTRUMENTO PARA VALIDACIÓN DE LA PROPUESTA

Estimado colega:

Se solicita su valiosa cooperación para evaluar la siguiente propuesta del proyecto de titulación: **Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal.**

Sus criterios son de suma importancia para la realización de este trabajo, por lo que se le pide brinde su cooperación contestando las preguntas que se realizan a continuación.

Datos informativos

Validado por:

Título obtenido
Master en Ciencia de Datos
Cédula de Identidad
0503727448
E- mail
gisselaguanoluisa7448@utc.edu.ec
Institución de Trabajo
Departamento de Seguridad y Defensa
Cargo
Docente software
Años de experiencia en el área
5 años

Instructivo:

- Responda cada criterio con la máxima sincera del caso;
- Revisar, observar y analizar la propuesta del proyecto de titulación; y,
- Coloque **una X** en cada indicador, tomando en cuenta que Muy adecuado equivale a 5, Bastante Adecuado equivale a 4, Adecuado equivale a 3, Poco Adecuado equivale a 2 e Inadecuado equivale a 1.

Tema:

Indicador	Descripción	Muy adecuado	Bastante Adecuado	Adecuado	Poco adecuado	Inadecuado
Impacto	El alcance que tendrá la propuesta y su representatividad en la generación de valor					X
Aplicabilidad	La capacidad de implementación de la propuesta considerando que los contenidos sean aplicables				X	
Conceptualización	La base de conceptos y teorías propias de la propuesta de manera sistémica y articulada					X
Actualidad	Los procedimientos actuales y los cambios científicos y tecnológicos considerados en la propuesta					X
Calidad Técnica	Los atributos cualitativos del contenido de la propuesta para satisfacer las expectativas de sus beneficiarios				X	
Factibilidad	El nivel de utilización de la propuesta por parte de la organización acorde a los recursos disponibles				X	
Pertinencia	La contundencia y conveniencia de la propuesta para solucionar el problema planteado.					X
Total					12	20

Observaciones:

Excelente trabajo.

Recomendaciones

Se recomienda en trabajos futuros implementar redes neuronales ya entrenadas con el fin de precisar los resultados en futuros entrenamientos de redes que beneficiaria en próximos proyectos similares.

Lugar, fecha de validación: 14 agosto 2025



Firma del especialista

INSTRUMENTO DE VALIDACIÓN

UNIVERSIDAD TECNOLÓGICA ISRAEL

ESCUELA DE POSGRADOS "ESPOG"

MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

INSTRUMENTO PARA VALIDACIÓN DE LA PROPUESTA

Estimado colega:

Se solicita su valiosa cooperación para evaluar la siguiente propuesta del proyecto de titulación: **Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal.**

Sus criterios son de suma importancia para la realización de este trabajo, por lo que se le pide brinde su cooperación contestando las preguntas que se realizan a continuación.

Datos informativos

Validado por: David Rivas Lalaleo

Título obtenido
Phd
Cédula de Identidad
1802445302
E- mail
drrivas@espe.edu.ec
Institución de Trabajo
ESPE
Cargo
Jefe de Comunicaciones
Años de experiencia en el área
15 años

Instructivo:

- Responda cada criterio con la máxima sincera del caso;
- Revisar, observar y analizar la propuesta del proyecto de titulación; y,
- Coloque **una X** en cada indicador, tomando en cuenta que Muy adecuado equivale a 5, Bastante Adecuado equivale a 4, Adecuado equivale a 3, Poco Adecuado equivale a 2 e Inadecuado equivale a 1.

Tema:

Indicador	Descripción	Muy adecuado	Bastante Adecuado	Adecuado	Poco adecuado	Inadecuado
Impacto	El alcance que tendrá la propuesta y su representatividad en la generación de valor					X
Aplicabilidad	La capacidad de implementación de la propuesta considerando que los contenidos sean aplicables				X	
Conceptualización	La base de conceptos y teorías propias de la propuesta de manera sistémica y articulada					X
Actualidad	Los procedimientos actuales y los cambios científicos y tecnológicos considerados en la propuesta					X
Calidad Técnica	Los atributos cualitativos del contenido de la propuesta para satisfacer las expectativas de sus beneficiarios					X
Factibilidad	El nivel de utilización de la propuesta por parte de la organización acorde a los recursos disponibles				X	
Pertinencia	La contundencia y conveniencia de la propuesta para solucionar el problema planteado.				X	
Total					12	20

Observaciones:

Excelente proyecto mejorar en la factibilidad.

Recomendaciones

Se recomienda utilizar otras redes neuronales para mejorar la predicción, redes con más entrenamiento que no sean con mayor número de capas.

Lugar, fecha de validación: 15 septiembre 2025



DAVID NAIMUNDO
RIVERA LAJALBO

Firma del especialista

INSTRUMENTO DE VALIDACIÓN

UNIVERSIDAD TECNOLÓGICA ISRAEL

ESCUELA DE POSGRADOS "ESPOG"

MAESTRÍA EN ELECTRÓNICA Y AUTOMATIZACIÓN

INSTRUMENTO PARA VALIDACIÓN DE LA PROPUESTA

Estimado colega:

Se solicita su valiosa cooperación para evaluar la siguiente propuesta del proyecto de titulación: **Sistema ambulatorio de detección de anomalías de signos vitales mediante IoT y red neuronal.**

Sus criterios son de suma importancia para la realización de este trabajo, por lo que se le pide brinde su cooperación contestando las preguntas que se realizan a continuación.

Datos Informativos

Validado por: Walter Javier Villarroel Herrera

Título obtenido
Ing. Electrónica e Instrumentación
Cédula de Identidad
0503108318
E- mail
Walterusbb@hotmail.com
Institución de Trabajo
Vgoldtrack
Cargo
Gerente General
Años de experiencia en el área
8 años

Instructivo:

- Responda cada criterio con la máxima sincera del caso;
- Revisar, observar y analizar la propuesta del proyecto de titulación; y,
- Coloque **una X** en cada indicador, tomando en cuenta que Muy adecuado equivale a 5, Bastante Adecuado equivale a 4, Adecuado equivale a 3, Poco Adecuado equivale a 2 e Inadecuado equivale a 1.

Tema:

Indicador	Descripción	Muy adecuado	Bastante Adecuado	Adecuado	Poco adecuado	Inadecuado
Impacto	El alcance que tendrá la propuesta y su representatividad en la generación de valor					X
Aplicabilidad	La capacidad de implementación de la propuesta considerando que los contenidos sean aplicables				X	
Conceptualización	La base de conceptos y teorías propias de la propuesta de manera sistémica y articulada					X
Actualidad	Los procedimientos actuales y los cambios científicos y tecnológicos considerados en la propuesta					X
Calidad Técnica	Los atributos cualitativos del contenido de la propuesta para satisfacer las expectativas de sus beneficiarios					X
Factibilidad	El nivel de utilización de la propuesta por parte de la organización acorde a los recursos disponibles				X	
Pertinencia	La contundencia y conveniencia de la propuesta para solucionar el problema planteado.					X
Total					12	25

Observaciones:

Se debe tener en cuenta tarjetas industriales para el proyecto.

Recomendaciones

Se recomienda que se pueda hacer una tarjeta propia para poder hacer producción nacional.

Lugar, fecha de validación: 15 agosto 2025



Firma del especialista