



**UNIVERSIDAD ESTATAL PENÍNSULA  
DE SANTA ELENA  
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES  
INSTITUTO DE POSTGRADO**

**TÍTULO**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO  
DOMICILIARIO PARA LA GESTIÓN INTEGRAL DE LA SALUD  
CARDIOVASCULAR**

**AUTOR**

**Alomoto Tomalá Mario Enrique**

**TRABAJO DE TITULACIÓN**

**Previo a la obtención del grado académico en  
MAGÍSTER EN ELECTRÓNICA Y AUTOMATIZACIÓN**

**TUTOR**

**Ing. Manuel Montaña Blacio. MSc.**

**Santa Elena, Ecuador**

**Año 2024**



**UNIVERSIDAD ESTATAL PENÍNSULA  
DE SANTA ELENA  
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES  
INSTITUTO DE POSTGRADO  
TRIBUNAL DE SUSTENTACIÓN**

---

**Ing. Alicia Andrade Vera, Mgtr.  
COORDINADORA DEL  
PROGRAMA**

---

**Ing. Manuel Montaña Blacio, MSc.  
TUTOR**

---

**Ing. Junior Figueroa Olmedo, MSc.  
DOCENTE  
ESPECIALISTA**

---

**Ing. Sendey Vera González, MSc.  
DOCENTE  
ESPECIALISTA**

---

**Abg. María Rivera, MSc.  
SECRETARIA GENERAL  
UPSE**



**UNIVERSIDAD ESTATAL PENÍNSULA  
DE SANTA ELENA  
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES  
INSTITUTO DE POSTGRADO**

**CERTIFICACIÓN**

Certifico que luego de haber dirigido científica y técnicamente el desarrollo y estructura final del trabajo, este cumple y se ajusta a los estándares académicos, razón por el cual apruebo en todas sus partes el presente trabajo de titulación que fue realizado en su totalidad por ALOMOTO TOMALÁ MARIO ENRIQUE, como requerimiento para la obtención del título de Magíster en Electrónica y Automatización.

**TUTOR**

---

**Ing. Manuel Montaña Blacio. MSc.**

Santa Elena, 21 de junio de 2024



**UNIVERSIDAD ESTATAL PENÍNSULA  
DE SANTA ELENA  
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES  
INSTITUTO DE POSTGRADO**

**DECLARACIÓN DE RESPONSABILIDAD**

**Yo, ALOMOTO TOMALÁ MARIO ENRIQUE**

**DECLARO QUE:**

El trabajo de Titulación, Diseño e implementación de un Sistema de Monitoreo Domiciliario para la Gestión Integral de la Salud Cardiovascular, previo a la obtención del título en Magíster en Electrónica y Automatización, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Santa Elena, 21 de junio de 2024

**EL AUTOR**

---

**Mario Enrique Alomoto Tomalá**



**UNIVERSIDAD ESTATAL PENÍNSULA  
DE SANTA ELENA  
FACULTAD DE CIENCIAS DE LA INGENIERÍA  
INSTITUTO DE POSTGRADO  
CERTIFICACIÓN DE ANTIPLAGIO**

Certifico que después de revisar el documento final del trabajo de titulación denominado Diseño e implementación de un Sistema de Monitoreo Domiciliario para la Gestión Integral de la Salud Cardiovascular, presentado por el estudiante, ALOMOTO TOMALÁ MARIO ENRIQUE fue enviado al Sistema Anti plagio COMPILATIO, presentando un porcentaje de similitud correspondiente al 4%, por lo que se aprueba el trabajo para que continúe con el proceso de titulación.

 CERTIFICADO DE ANÁLISIS  
magister

**TRABAJO DE TITULACION - MARIO  
ALOMOTO TOMALA**

**4%**  
Textos sospechosos

**4% Similitudes**  
0% similitudes entre comillas  
0% entre las fuentes mencionadas  
0% Idiomas no reconocidos

Nombre del documento: TRABAJO DE TITULACION - MARIO ALOMOTO TOMALA.pdf ID del documento: 8ba4953ebf83f18c6a23c9f2f436f8aef0ea9fa0 Tamaño del documento original: 1,63 MB	Depositante: Manuel Asdrual Montaña Blacio Fecha de depósito: 21/6/2024 Tipo de carga: interfase fecha de fin de análisis: 21/6/2024	Número de palabras: 11.687 Número de caracteres: 91.226
--	---	--

**TUTOR**

---

**Ing. Manuel Montaña Blacio. MSc.**



**UNIVERSIDAD ESTATAL PENÍNSULA  
DE SANTA ELENA  
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES  
INSTITUTO DE POSTGRADO**

**AUTORIZACIÓN**

**Yo, ALOMOTO TOMALÁ MARIO ENRIQUE**

Autorizo a la Universidad Estatal Península de Santa Elena, para que haga de este trabajo de titulación o parte de él, un documento disponible para su lectura consulta y procesos de investigación, según las normas de la Institución.

Cedo los derechos en línea patrimoniales de mi trabajo de examen complejo con fines de difusión pública, además apruebo la reproducción de este trabajo de examen complejo dentro de las regulaciones de la Universidad, siempre y cuando esta reproducción no suponga una ganancia económica y se realice respetando mis derechos de autor.

Santa Elena, 21 de junio de 2024

**EL AUTOR**

---

**Mario Enrique Alomoto Tomalá**

## AGRADECIMIENTO

A Dios, por ser mi guía y fuente de fortaleza, por darme la sabiduría y la perseverancia necesarias para completar este trabajo. Sin su presencia y bendiciones, este logro no habría sido posible.

A mis padres, por su amor incondicional y su apoyo constante en cada paso de este camino. Su sacrificio y confianza en mí han sido fundamentales para alcanzar esta meta.

A mi futura esposa Angie, por su amor incondicional, su paciencia y su apoyo constante. Tus palabras de aliento y comprensión han sido una fuente de inspiración en los momentos más difíciles. Gracias por estar siempre a mi lado y por creer en mí.

A mis profesores y mentores, por su guía y sabiduría, que han sido esenciales en mi desarrollo académico y personal.

A mis amigos y compañeros, por su camaradería y ánimo en los momentos difíciles, compartiendo conmigo las alegrías y los desafíos de esta travesía.

Finalmente, a todas aquellas personas que, de una u otra forma, han contribuido a mi formación y crecimiento, gracias por ser parte de esta etapa tan importante de mi vida

*Mario Enrique, Alomoto Tomalá*

# DEDICATORIA

A mi familia, en especial a mi amada mamá por ser la motivación de realizar este trabajo de titulación.

A mi futura esposa por motivarme a nuevos desafíos y por creer siempre en mí.

*Mario Enrique, Alomoto Tomalá*

# ÍNDICE GENERAL

<b>TRIBUNAL DE SUSTENTACIÓN .....</b>	<b>II</b>
<b>CERTIFICACIÓN.....</b>	<b>III</b>
<b>DECLARACIÓN DE RESPONSABILIDAD .....</b>	<b>IV</b>
<b>CERTIFICACIÓN DE ANTIPLAGIO .....</b>	<b>V</b>
<b>AUTORIZACIÓN .....</b>	<b>VI</b>
<b>AGRADECIMIENTO .....</b>	<b>VII</b>
<b>DEDICATORIA.....</b>	<b>VIII</b>
<b>ÍNDICE GENERAL .....</b>	<b>IX</b>
<b>ÍNDICE DE TABLAS .....</b>	<b>XII</b>
<b>ÍNDICE DE FIGURAS .....</b>	<b>XIII</b>
<b>ANEXOS.....</b>	<b>XIV</b>
<b>RESUMEN .....</b>	<b>XV</b>
<b>ABSTRACT.....</b>	<b>XVI</b>
<b>INTRODUCCIÓN .....</b>	<b>2</b>
<b>DESARROLLO .....</b>	<b>5</b>
Metodología .....	5
Estado del arte .....	6
Materiales y métodos .....	8
Software .....	8
Tago IO – Plataforma IoT.....	8
Librería MAX30100_PulseOximeter .....	8
Hardware.....	9
Unidad de control Xiao ESP32C3 .....	9

Sensor de temperatura MLX90614.....	9
Modulo frecuencia cardiaca y oxígeno en la sangre.....	10
Batería de litio 3.7V/500mAh.....	11
Propuesta.....	13
Diseño electrónico.....	15
Diagrama de flujo.....	16
Diseño de la carcasa en 3D del prototipo.....	17
Implementación del sistema de monitoreo de la salud cardiovascular.....	18
Configuración de Alertas en la plataforma TagoIO.....	19
Aplicación de Modelos de Maching Learning para predicción de alteraciones en las variables de salud cardiovascular.....	20
<b>RESULTADOS .....</b>	<b>21</b>
Análisis de datos de frecuencia cardiaca.....	21
Análisis de datos de oxígeno en la sangre.....	23
Análisis de datos de temperatura corporal.....	25
Autonomía del dispositivo.....	27
Interpretación de histogramas con 200 datos.....	28
Interpretación de histogramas con 464 datos.....	29
Modelo Maching Learning: correlación entre variables.....	30
Entrenamiento 1.....	30
Entrenamiento 2.....	31
Modelo Maching Learning: Modelo de regresión utilizando método de permutación.....	32
Entrenamiento 1.....	32
Entrenamiento 2.....	33
Modelo Maching Learning: Modelo LSTM de redes neuronales.....	34
Entrenamiento 1.....	34
Entrenamiento 2.....	35

Análisis de RMSE (Raíz del error cuadrático medio) .....	36
Análisis de MAE (Error absoluto medio) .....	36
<b>CONCLUSIONES .....</b>	<b>37</b>
<b>RECOMENDACIONES .....</b>	<b>38</b>
<b>REFERENCIAS.....</b>	<b>39</b>
<b>ANEXOS.....</b>	<b>43</b>

## ÍNDICE DE TABLAS

<b>Tabla 1</b>	Características del smartwatch Xiaomi S1 PRO.....	6
<b>Tabla 2</b>	Características principales de la unidad de control ESP32C3 .....	9
<b>Tabla 3</b>	Características principales del Sensor MLX90614.....	10
<b>Tabla 4</b>	Características principales del sensor MAX30100.....	11
<b>Tabla 5</b>	Conexiones del sensor MAX30100 .....	15
<b>Tabla 6</b>	Conexiones del sensor MLX90614.....	16
<b>Tabla 7</b>	Consumo total de corriente del prototipo propuesto.....	28

## ÍNDICE DE FIGURAS

<b>Figura 1</b> Batería de litio 3.7V 500mAh .....	12
<b>Figura 2</b> Arquitectura propuesta para el sistema de monitoreo de la salud cardiovascular .....	13
<b>Figura 3</b> Interfaz de usuario del sistema de monitoreo de la salud cardiovascular .....	14
<b>Figura 4</b> Diagrama electrónico .....	15
<b>Figura 5</b> Diagrama de flujo del funcionamiento del prototipo .....	17
<b>Figura 6</b> Dimensiones de la carcasa del prototipo .....	18
<b>Figura 7</b> Implementación del prototipo .....	19
<b>Figura 8</b> Configuración de alerta temprana en TagoIO .....	20
<b>Figura 9</b> Gráfico de dispersión de FC en una persona sin hipertensión .....	21
<b>Figura 10</b> Gráfico de dispersión de FC en una persona hipertensa .....	22
<b>Figura 11</b> Gráfico de dispersión de SPO2 en una persona sin hipertensión .....	24
<b>Figura 12</b> Gráfico de dispersión de oxígeno en la sangre en una persona hipertensa ...	25
<b>Figura 13</b> Gráfico de dispersión de la temperatura en una persona sin hipertensión ....	26
<b>Figura 14</b> Gráfico de dispersión de la temperatura en una persona hipertensa .....	27
<b>Figura 15</b> Histograma de variables con una muestra de 200 datos.....	29
<b>Figura 16</b> Histograma de variable con una muestra de 464 datos .....	30
<b>Figura 17</b> Gráficos la matriz de correlación en una muestra de 200 datos.....	31
<b>Figura 18</b> Gráficos la matriz de correlación en una muestra de 464 datos.....	32
<b>Figura 19</b> Gráfico de permutación en una muestra de 200 datos .....	33
<b>Figura 20</b> Gráfico de permutación en una muestra de 464 datos .....	34
<b>Figura 21</b> Gráfico obtenido del modelo LSTM con una muestra de 200 datos.....	35
<b>Figura 22</b> Gráfico obtenido del modelo LSTM con una muestra de 464 datos.....	35

## ANEXOS

<b>Anexo 1</b> Código en Google Colab de histograma y correlación con 200 datos de muestra.....	43
<b>Anexo 2</b> Código en google colab de histograma y correlación con 464 datos de muestra .....	44
<b>Anexo 3</b> Modelo de regresión utilizando el método de permutación con 200 datos.....	45
<b>Anexo 4</b> Modelo de regresión utilizando el método de permutación con 646 datos.....	46
<b>Anexo 5</b> Modelo LSTM con 200 datos .....	47
<b>Anexo 6</b> Modelo LSTM con 646 datos .....	49
<b>Anexo 7</b> Código de dispositivo desarrollado en la IDE Arduino .....	54

## RESUMEN

Los dispositivos y sistemas de monitoreo son herramientas indispensables en pacientes con problemas en la salud cardiovascular, herramientas que son operadas por personal capacitado en instituciones de salud, y en otros casos son accesibles pero operadas por personal certificado. El objetivo de este trabajo es presentar un dispositivo de monitoreo y control basado en internet de las cosas (IoT) para pacientes con problemas de salud cardiovascular. El dispositivo toma lecturas de frecuencia cardiaca, oxígeno en la sangre y la temperatura corporal y posteriormente son enviadas a través de Internet a una plataforma IoT para el monitoreo por parte del médico tratante o tutor. La interfaz permite monitorear los datos recolectados y en caso de obtener valores inusuales, se presenta una notificación en la interfaz para atención oportuna al paciente. Se diseñó e imprimió en 3D una carcasa ergonómica para protección de los componentes electrónicos y para la implementación en el paciente. Los resultados obtenidos se compararon con un reloj inteligente de alto valor comercial, los cuales indican que la confiabilidad con respecto a la frecuencia cardiaca es del 85%, 90% en el oxígeno en la sangre y 92% en la temperatura.

**Palabras clave:** Monitoreo domiciliario, dispositivo portátil IOT, salud cardiovascular.

## **ABSTRACT**

Monitoring devices and systems are essential tools for patients with cardiovascular health problems, tools that are operated by trained personnel in health institutions, and in other cases they are accessible but operated by certified personnel. The objective of this work is to present a monitoring and control device based on the Internet of Things (IoT) for patients with cardiovascular health problems. The device takes readings of heart rate, blood oxygen and body temperature and are subsequently sent over the Internet to an IoT platform for monitoring by the treating doctor or guardian. The interface allows you to monitor the collected data and if unusual values are obtained, a notification is presented in the interface for timely patient care. An ergonomic housing was designed and 3D printed to protect the electronic components and for implementation on the patient. The results obtained were compared with a smart watch of high commercial value, which indicate that the reliability with respect to heart rate is 85%, 90% in blood oxygen and 92% in temperature.

**Keywords:** Home monitoring, IOT wearable device, cardiovascular health.

## INTRODUCCIÓN

Las enfermedades cardiovasculares (ECV) representan una de las principales causas de morbilidad y mortalidad a nivel mundial. De acuerdo a las estadísticas realizadas por la Organización Panamericana de la Salud (OPS): 2 millones de personas murieron a causa de las enfermedades cardiovasculares en el año 2019 (OPS, 2019).

Según una encuesta STEPS realizada por el Ministerio de Salud (MSP) y el Instituto Nacional de Estadísticas y Censos (INEC) en el año 2018, los datos en relación a la salud cardiovascular de personas entre 18 y 69 años, se encontró que el 45.2% registraba presión arterial elevada sin diagnóstico, el 12.6% registraba presión elevada con diagnóstico pero sin ningún tipo de tratamiento, el 16.2% presentaba presión elevada diagnosticada y con tratamiento, sin embargo se mantenía descontrolada; y el 26% presentaba presión arterial elevada con tratamiento y controlada (Barrenech, 2023).

A pesar de los avances tecnológicos, todavía existen limitaciones en el acceso a la atención médica continua y la detección temprana de problemas de salud. El seguimiento actual de la salud, especialmente para pacientes con enfermedades crónicas o en riesgo, presenta limitaciones significativas que afectan la calidad de la atención médica desde agendamiento de turnos hasta la escasez de recursos humanos en el ámbito de la salud (Naranjo, 2016).

Dentro de las limitaciones encontramos que las visitas al médico y las pruebas de laboratorio, se realizan con poca frecuencia, el acceso a la atención médica especializada, en especial a cardiología, es restringido para algunos pacientes, especialmente para aquellos que tienen movilidad reducida. Así mismo los profesionales de la salud no siempre tienen acceso a datos actualizados de sus pacientes, lo que dificulta la toma de decisiones informadas sobre el tratamiento (Witt et al., 2023).

Como objetivo principal de este proyecto es desarrollar una solución tecnológica que permita a personas adultas mayores que sufran de enfermedades cardiovasculares, realizar un monitoreo y control continuo de variables tales como frecuencia cardiaca, oxígeno en la sangre y temperatura, de tal manera que si se presente una alteración permita realizar la asistencia temprana y oportuna.

El dispositivo electrónico consta de una batería de litio recargable de 3.7V como voltaje nominal y hasta 4.2V como máximo, un conector USB de tipo C para realizar la carga; un indicador led que permite al usuario mostrar que el dispositivo inició su software, se configuraron correctamente los sensores, se conectó a la red WiFi del hogar, posteriormente la conexión a la plataforma IoT Tago IO el mismo que aloja los datos en la nube.

El dispositivo electrónico está ideado para el uso en personas adultas mayores que tengan una edad de entre 55 a 60 años. Además, que estas personas sufran de enfermedades cardiovasculares, principalmente tengan un antecedente de derrame cerebral de tipo isquémico leve y que sufran de alteraciones continuas en su presión arterial a pesar de tener su tratamiento y se encuentren medicados.

Este documento está compuesto por el marco teórico el cual abarca la selección de los componentes electrónicos utilizados para el hardware, los programas utilizados para el desarrollo del algoritmo, diseño del esquema electrónico y diseño de la estructura física que protege el dispositivo; además, se presenta la metodología utilizada para comprobar la fiabilidad del funcionamiento del dispositivo, los resultados obtenidos, conclusiones y las respectivas recomendaciones.

## **OBJETIVO GENERAL**

Desarrollar un sistema de monitoreo domiciliario que utilice microcontroladores, sensores e internet de las cosas (IoT), que evalúe la salud cardiovascular y proporcione información crítica sobre posibles riesgos mediante alertas tempranas.

## **OBJETIVOS ESPECÍFICOS**

- Desarrollar un dispositivo electrónico basado en sensores que permitan medir variables relevantes para la salud cardiovascular.
- Integrar una plataforma de internet de las cosas (IoT) al dispositivo electrónico para visualizar los datos de la salud cardiovascular.
- Desarrollar un modelo predictivo de acuerdo a los valores de sensores que integra el dispositivo electrónico, para monitorear las alteraciones en la salud cardiovascular.

# DESARROLLO

## Metodología

La propuesta tecnológica presentada fue desarrollada en el barrio 28 de mayo y está dirigida a personas con enfermedades cardiovasculares centradas en la alteración de la presión arterial. Este proyecto adopta un enfoque experimental cuantitativo, que se fundamenta en el diseño e implementación de un prototipo IoT para monitorear variables tales como: frecuencia cardíaca, los niveles de oxígeno en la sangre y la temperatura corporal. Para llevar a cabo esta propuesta, se ha diseñado un modelo basado en fases, siguiendo la arquitectura propuesta.

La primera etapa de esta propuesta consistió en el desarrollo del prototipo electrónico basado en el Internet de las Cosas (IoT), junto con la integración de componentes electrónicos de hardware y software de código abierto. En la segunda etapa, se procedió a la recolección de datos de frecuencia cardíaca y niveles de oxígeno en la sangre, obtenidos a través del sensor MAX30100, los cuales fueron registrados en unidades de latidos por minuto y porcentajes respectivamente. Para la medición de la temperatura corporal en grados Celsius, se empleó el sensor MLX90614.

Como unidad de control del sistema electrónico se empleó la placa de desarrollo Xiao ESP32C3, el cual cumple la función de procesamiento de los datos recolectados de los sensores mencionados con antelación, el núcleo central de la propuesta se basó en desarrollar un modelamiento mediante los datos recolectados, de tal manera que el sistema predice alteraciones en la salud cardiovascular y posteriormente enviar alertas para brindar una atención oportuna mediante tecnología IoT en tiempo real.

La tercera fase se centra en el diseño y la impresión 3D de una carcasa destinada a integrar los componentes electrónicos de manera que el dispositivo resulte compacto, ergonómico y no invasivo para el usuario. La consideración de aspectos no invasivos es crucial, especialmente para dispositivos que requieren un uso constante, ya que promueve la aceptación y satisfacción del usuario, incrementando la probabilidad de un uso continuo y efectivo.

La cuarta fase se enfoca en la comparación y validación de datos del prototipo con los datos de un reloj inteligente referentes a las variables de frecuencia cardíaca, oxígeno en

la sangre y la temperatura corporal. Para este procedimiento se optó por usar el smartwatch Xiaomi S1 pro, el mismo que cumple la conformidad de la Unión Europea (UE). A continuación, en la tabla 1 muestra las características principales como es el nombre del fabricante y modelo del reloj (Xiaomi, 2024).

**Tabla 1**

*Características del smartwatch Xiaomi S1 PRO*

Características	Valores
Producto	Reloj inteligente
Nombre	Xiaomi S1 PRO
Modelo	M2112W1

*Nota.* Los datos presentados en esta tabla fueron adaptados del manual proporcionado por el fabricante Xiaomi Communications Co., Ltd.

La fase final consistió en la presentación de datos al usuario tutor, lo cual permite monitorear el estado de las variables definidas en la propuesta. Además, se integró un modelo que detecta variaciones en dichas variables, proporcionando alertas tempranas y permitiendo una asistencia oportuna.

**Estado del arte**

(Aditya, 2020) en su artículo científico nos presenta un dispositivo para medir el nivel de oxígeno en sangre de forma no invasiva, utilizando el sensor MAX30100 y un microcontrolador Arduino Uno. El dispositivo incluye una alarma que se activa cuando los niveles de oxígeno caen por debajo del 95%, lo que lo convierte en una herramienta valiosa para la monitorización de la salud. Además, integra una alarma para que el dispositivo alerte al usuario cuando los niveles de SpO2 estén por debajo de los niveles normales, lo que permite una intervención rápida en caso de hipoxemia. Adicionalmente, el dispositivo cuenta con una pantalla LCD que muestra los resultados de las mediciones de SpO2 de manera clara y sencilla. Las pruebas realizadas demostraron que las mediciones del dispositivo son comparables a las de herramientas calibradas, con un error medio de tan solo 0,0123%.

(Garces et al., 2021) en “Plataforma IoT de bajo-coste para aplicaciones de telemedicina”, presenta un sistema de telemedicina domiciliaria el cual consiste en la monitorización de pacientes a distancia. El sistema utiliza dispositivos IoT (Internet de las Cosas) y la nube para recopilar, analizar y compartir datos de salud de forma segura y eficiente. Los usuarios portan una pulsera económica que mide su frecuencia cardíaca (BPM) y saturación de oxígeno (SpO2). Estos datos se transmiten por Bluetooth a una aplicación móvil que se ejecuta en un Android TV. La aplicación envía los datos a la nube, donde son analizados por profesionales de la salud. Basándose en el análisis de los datos, los profesionales pueden emitir prescripciones médicas que se envían al usuario a través de la aplicación. El sistema también proporciona a los profesionales, herramientas para realizar un seguimiento del progreso del paciente y gestionar su atención médica. Una encuesta realizada a profesionales de la salud arrojó resultados positivos. La mayoría de los encuestados (88,5%) expresaron su disposición a implementar el sistema en su práctica, reconociendo la importancia de los datos recopilados (92,4%) y la utilidad de las herramientas del sistema para la atención médica (76,9%).

(Manzano & Bejarano, 2021) nos presenta su propuesta que consiste en el desarrollo de un sistema de monitorización, el cual está especialmente diseñado para el cuidado de pacientes en su domicilio, brindando una solución compacta y económica que ofrece información médica oportuna. El sistema se ha diseñado con la perspectiva de incorporar en el futuro tecnologías de computación en la nube e Internet de las Cosas, permitiendo el monitoreo remoto de los datos del paciente. El artículo describe en detalle el diseño e implementación de la tarjeta principal del sistema, la selección de los sensores biomédicos, la elaboración de los soportes y la programación del sistema.

(Vela et al., 2024) en su trabajo de investigación nos presenta un sistema de monitoreo automatizado que garantice el acceso remoto y la captura eficiente de la información. En esta investigación se utilizaron enfoques cualitativos y cuantitativos. Se realizaron pruebas de usuario para obtener opiniones sobre la interfaz propuesta y se empleó un método comparativo para evaluar la efectividad del sistema en comparación con equipos disponibles en el mercado. La fase de transmisión de información envía los datos obtenidos de los sensores a la plataforma de IoT ThingSpeak y, paralelamente, utiliza una conexión de área local a través de la plataforma VNC Viewer para asegurar una doble comunicación en caso de que los servidores de ThingSpeak fallen. Aunque el sistema

propuesto puede medir parámetros básicos como la frecuencia cardíaca, temperatura y electrocardiograma (ECG), no es adecuado para proporcionar un análisis médico definitivo. Sin embargo, es útil para obtener datos básicos en entornos educativos, como en el Instituto Superior Tecnológico Luis Tello en la provincia de Esmeraldas, Ecuador.

## **Materiales y métodos**

### **Software**

#### **Tago IO – Plataforma IoT**

TagoIO es una plataforma de Internet de las Cosas (IoT) que proporciona herramientas para la gestión y el análisis de datos recolectados por dispositivos IoT. Permite a los usuarios conectar, visualizar, almacenar y analizar datos de una manera. TagoIO se distingue por ofrecer una variedad de funcionalidades que facilitan el desarrollo y la implementación de soluciones IoT (TagoIO, 2023).

La presente propuesta se beneficia significativamente de las características ofrecidas por la plataforma al permitir el consumo de sus servicios a través de solicitudes HTTP. Esto facilita que desde la unidad de control se configuren las transmisiones de datos hacia la plataforma IoT mediante la escritura de líneas de código.

#### **Librería MAX30100\_PulseOximeter**

Usar simultáneamente los sensores oxímetro Max30100 y de temperatura MLX90614 presenta un desafío debido a que ambos emplean el mismo canal de comunicación I2C para transmitir sus valores a la unidad de control. Este conflicto se puede resolver ajustando la frecuencia de reloj del sensor de oximetría a través de los parámetros de su librería en la IDE de Arduino. Aunque la frecuencia de reloj estándar es de 100 kHz, en esta propuesta se recomienda incrementarla a 400 kHz para mejorar la sincronización y evitar colisiones en el bus de datos. Adicionalmente, es crucial definir correctamente las direcciones I2C de cada sensor en el código. Para el sensor de oximetría, la dirección es 0x57, mientras que para el sensor de temperatura es 0x5A. Al incluir las librerías correspondientes de ambos sensores, se deben especificar estas direcciones para asegurar que la unidad de control pueda distinguir y comunicarse adecuadamente con cada dispositivo (Forum Arduino, 2022).

## Hardware

### Unidad de control Xiao ESP32C3

La placa de desarrollo Xiao ESP32C3 es una opción ideal para proyectos que requieren un diseño compacto debido a su tamaño reducido y su alta funcionalidad. Esta placa admite múltiples protocolos de comunicación, incluyendo serial, SPI e I2C. El protocolo I2C, en particular, es especialmente ideal para la comunicación con los sensores integrados en el dispositivo final, ya que permite la recepción de datos de los sensores en la placa de desarrollo para su posterior procesamiento. En la Tabla 1 se detallan las características principales del módulo de control seleccionado para esta propuesta, proporcionando una visión completa de sus capacidades y ventajas técnicas (Espressif Systems, 2024)

**Tabla 2**

*Características principales de la unidad de control ESP32C3*

Característica	Valores
Imagen de referencia	
Modelo	Xiao ESP32C3
Voltaje de entrada	3.3 [v]
Voltaje de entrada máx.	5 [v]
Conectividad WIFI	Si
Protocolos de comunicación	UART, SPI, I2C.
Dimensiones	21 x 17.5 [mm]
Consumo de corriente	[<75 mA]

*Nota.* Los datos presentados en esta tabla fueron adaptados del datasheet proporcionado por el fabricante Seeed Studio.

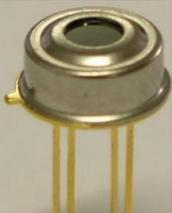
### Sensor de temperatura MLX90614

Este sensor es un termómetro infrarrojo que permite tomar lecturas de temperatura sin contacto. El MLX90614 puede medir temperaturas en un rango de -20 a 120 grados Celsius, y ofrece una alta precisión con una resolución de salida de 0.14 grados Celsius.

En la Tabla 2 se presentan las características principales del sensor de temperatura corporal MLX90614 (Melexis Inspired Engineering, 2019).

**Tabla 3**

*Características principales del Sensor MLX90614*

Características	Valores
Imagen de referencia	
Modelo	MLX90614
Voltaje operativo	3.3 [v]
Bajo consumo de corriente	170 [uA]
Rango de temperatura	-40 a 85 °C
Protocolo de comunicación	I2C

*Nota.* Los datos presentados en esta tabla fueron adaptados del datasheet proporcionado por el fabricante Melexis Inspired Engineering.

### **Modulo frecuencia cardiaca y oxígeno en la sangre**

El uso del MAX30100 en esta propuesta tecnológica se debe a su capacidad de proporcionar datos para ambos parámetros vitales. Los dos LEDs, uno de luz roja y otro de luz infrarroja, trabajan junto al fotodetector para medir la cantidad de luz absorbida por el flujo sanguíneo, lo que permite calcular tanto la frecuencia cardíaca como el nivel de oxígeno en la sangre. Esta funcionalidad dual es esencial en la integración de la presente propuesta por sus dimensiones físicas. En la tabla3, se muestran las características principales del sensor MAX30100 (Maxim Integrated, 2014).

**Tabla 4***Características principales del sensor MAX30100*

Características	Valores
Imagen de referencia	
Modelo	MAX30100
Voltaje operativo	5 [v]
Regulador de voltaje	3.3 [v] y 1.8 [v]
Consumo de corriente	1.2 [mA]
Protocolo de Comunicación	I2C
Temperatura de trabajo	-40°C a +85°C
Led rojo (E)	660 nm
Led infrarrojo (D)	920 nm
Dimensiones	14mm x 13mm x 14 mm

*Nota.* Los datos presentados en esta tabla fueron adaptados del datasheet proporcionado por el fabricante Analog Device.

#### **Batería de litio 3.7V/500mAh**

La implementación de dispositivos electrónicos destinados al monitoreo continuo de parámetros referentes a la salud cardiovascular, requiere fuentes de energía confiables y eficientes que permitan un funcionamiento ininterrumpido durante períodos prolongados. En este contexto, se emplea una batería de litio de 3.7V con una capacidad de 500mAh (Gang, 2024) como la fuente de alimentación primaria para nuestro dispositivo de

monitoreo de frecuencia cardíaca, oxígeno en la sangre y temperatura corporal. En la figura, se muestra la estructura física de la batería de litio.

### **Figura 1**

*Batería de litio 3.7V 500mAh*



*Nota.* Adaptado de Batería de litio 3.7V 500 mAh [Imagen], por Megatronica, 2024, tienda (<https://megatronica.cc/producto/bateria-de-lipo-3-7v-500mah-litio-25c/>).

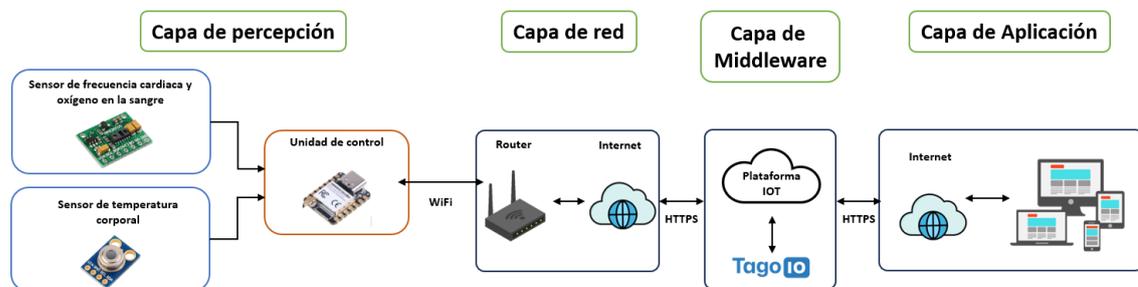
## Propuesta

El análisis de arquitecturas que incorporan el Internet de las cosas (IoT) implica la evaluación de diversos modelos que pueden constar de múltiples capas. Se sostiene que la robustez de una arquitectura se relaciona directamente con la cantidad de capas que la conforman. Este enfoque se basa en varios argumentos sólidos. Primero, la inclusión de múltiples capas permite una distribución eficiente de las responsabilidades y funciones dentro del sistema. Cada capa puede encargarse de tareas específicas, como la adquisición de datos, el procesamiento, la comunicación y la gestión de la seguridad. Esta segmentación reduce la complejidad de cada capa individual, lo que facilita la identificación y resolución de problemas (Buitrón, 2024).

La figura 2 ilustra la arquitectura del sistema de monitoreo de la salud cardiovascular, que se caracteriza por su enfoque de múltiples capas. Estas capas desempeñan roles específicos y están diseñadas para aprovechar las capacidades del Internet de las cosas (IoT). El modelo constituye cuatro capas: capa de percepción, capa de red, capa de middleware y capa de aplicación (Montaño-Blacio et al., 2023).

### Figura 2

*Arquitectura propuesta para el sistema de monitoreo de la salud cardiovascular*



La capa de percepción en el sistema de monitoreo de la salud cardiovascular tiene la función de detectar y medir los parámetros vitales, como la frecuencia cardíaca, los niveles de oxígeno en la sangre y la temperatura corporal. Una vez obtenidos estos datos, son transmitidos a la unidad de control. Esta unidad, a su vez, recibe y procesa la información con el objetivo de prepararla para su posterior tratamiento en la siguiente capa definida dentro de la arquitectura del sistema.

La capa de red posibilita la conexión inicial del sistema de monitoreo a una red inalámbrica, utilizando WiFi, lo que garantiza que el dispositivo tenga acceso a internet desde el principio. Este procedimiento es fundamental para la integración de la capa middleware debido a que la continuidad de operación del sistema de monitoreo de la salud cardiovascular depende directamente de la capa de red. Sin una conexión a la red que ofrezca acceso a internet, el intercambio de datos para el monitoreo no sería viable (Montaño et al., 2023).

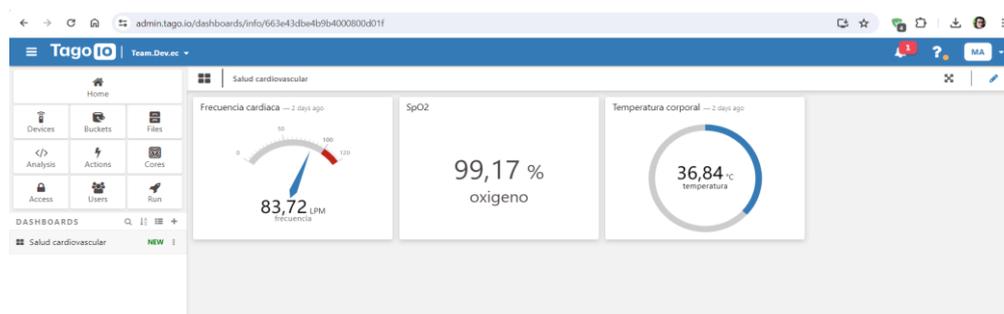
La implementación de HTTPS dentro de la capa middleware para la comunicación con la plataforma Tago IO es una decisión crucial en términos de seguridad y protección de datos. HTTPS, o Protocolo de Transferencia de Hipertexto Seguro, garantiza que la comunicación entre el servidor y el cliente esté cifrada, lo que previene la interceptación no autorizada de información.

En cuanto al formato de los datos enviados, el uso del método POST junto con el formato JSON proporciona facilidad de interpretación. El formato JSON (JavaScript Object Notation) es ampliamente utilizado debido a su simplicidad y legibilidad, lo que facilita el intercambio de datos entre sistemas heterogéneos. Al utilizar el método POST, los datos se envían de manera segura al servidor sin que queden expuestos en la URL, lo que es especialmente importante en cuanto a la seguridad de los datos.

Finalmente, la capa de aplicación propone una interfaz gráfica para el monitoreo de acuerdo a los sensores integrados al sistema de monitoreo de la salud cardiovascular. La interfaz presenta los datos que son enviados por el dispositivo para el respectivo monitoreo tal como se presenta en la figura 3.

### Figura 3

#### *Interfaz de usuario del sistema de monitoreo de la salud cardiovascular*

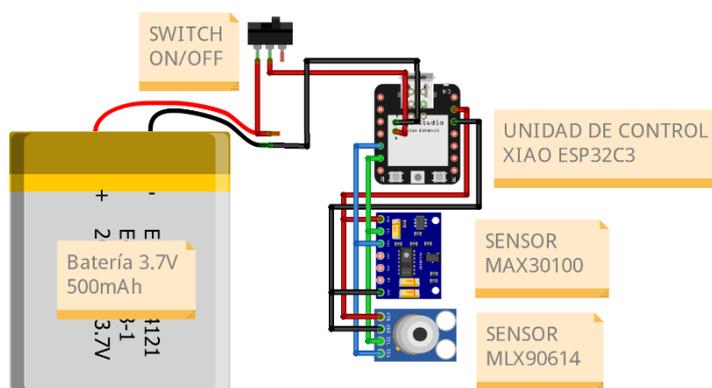


## Diseño electrónico

La Figura 4 presenta el diagrama electrónico del prototipo realizado en el software Fritzing (Fritzing, s. f.). El prototipo integra un sensor de temperatura, un sensor de frecuencia cardíaca y un sensor de oxígeno en la sangre, todos integrados con la placa de desarrollo Xiao ESP32C3 y una batería de litio de 3.7V. Es fundamental destacar que la unidad de control ESP32C3 no solo proporciona terminales específicos para la conexión de la batería de litio, sino que también permite la carga controlada de la batería a través de su puerto USB tipo C. Esta funcionalidad integrada simplifica considerablemente el diseño del sistema, ya que elimina la necesidad de componentes adicionales para gestionar la carga de la batería.

**Figura 4**

*Diagrama electrónico*



En las tablas 5 y 6 se detallan los pines de conexión para los sensores de frecuencia cardíaca, oxígeno en la sangre y temperatura corporal. Esta información es crucial para asegurar una comprensión clara y precisa de las conexiones necesarias para el funcionamiento óptimo de estos sensores.

**Tabla 5**

*Conexiones del sensor MAX30100*

Sensor MAX30100	Placa Xiao ESP32C3
VCC	3.3 V
SCL	D5
SDA	D4

INT	-
IRD	-
RD	-
GND	GND

**Tabla 6**

*Conexiones del sensor MLX90614*

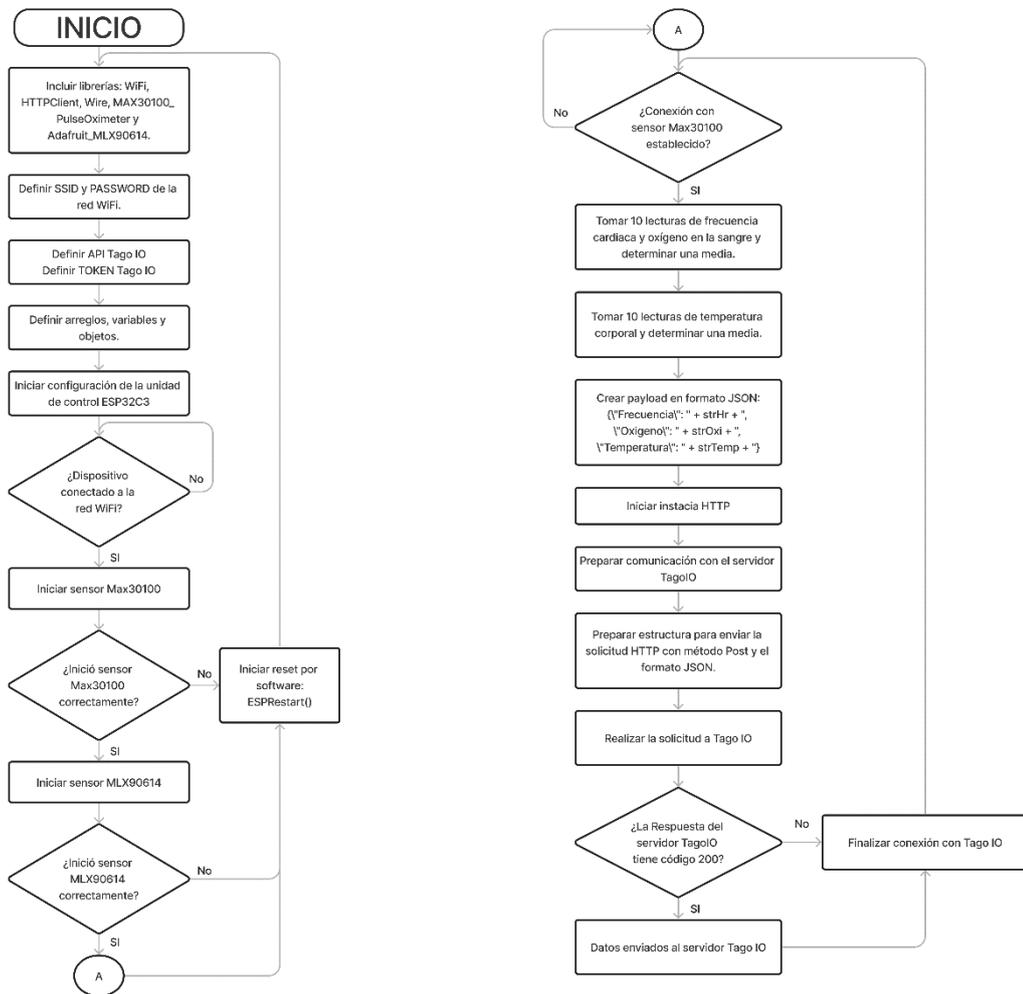
Sensor MLX90614	Placa Xiao ESP32C3
VIN	3.3 V
GND	GND
SCL	D5
SDA	D4

### **Diagrama de flujo**

El diagrama de flujo de la interacción de la unidad de control para la obtención de datos de los sensores y el envío de información a la plataforma IoT se muestra en la figura 5. El proceso comienza con la conexión a una red WiFi con acceso a internet, seguido de la configuración de los sensores para la lectura de datos y la conexión con la plataforma Tago IO. La conexión con la plataforma IoT sigue una secuencia ordenada: establecimiento de la conexión, estructuración del envío de la información y desconexión de la plataforma.

**Figura 5**

*Diagrama de flujo del funcionamiento del prototipo*



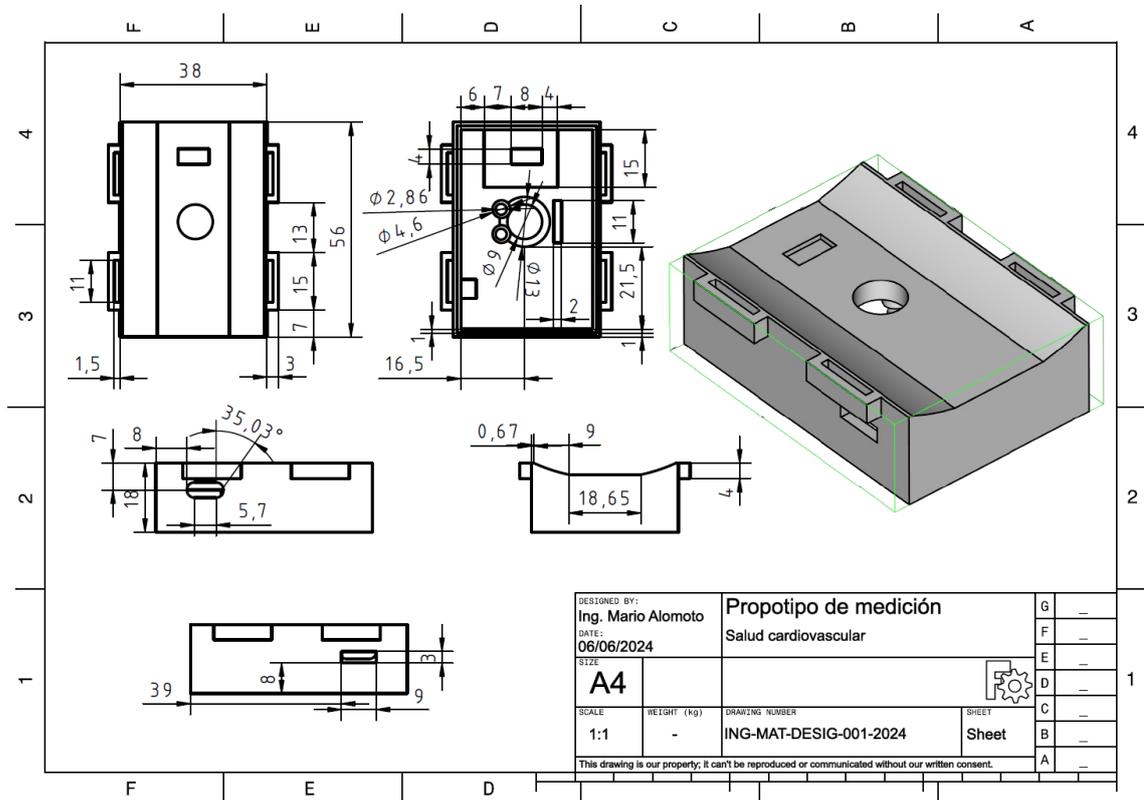
### Diseño de la carcasa en 3D del prototipo

La Figura 6 muestra el diseño en 3D de la carcasa del prototipo, que incorpora diversas características esenciales para su funcionalidad, usabilidad y respectivas dimensiones. Una de las características más destacadas es la ranura ubicada en la parte frontal, diseñada para facilitar la conexión del puerto USB tipo C a la unidad de control. Esta ranura permite tanto la programación del dispositivo como la carga controlada de la batería de litio, asegurando un acceso fácil y conveniente sin necesidad de desmontar el prototipo. Además, el diseño de la carcasa incluye ranuras específicamente posicionadas en la base para alojar los diferentes sensores: de temperatura, de frecuencia cardíaca y oxígeno en

la sangre. Estas ranuras no solo permiten una instalación segura y estable de los sensores, sino que también aseguran una exposición adecuada para la recolección de datos.

**Figura 6**

*Dimensiones de la carcasa del prototipo*



### Implementación del sistema de monitoreo de la salud cardiovascular

En la figura 7 se presenta la implementación del prototipo para el monitoreo de la salud cardiovascular en una paciente de 59 años de edad, de sexo femenino, con diagnóstico de hipertensión. El proceso de implementación implica la colocación del prototipo en el dedo índice de la mano derecha, seleccionando esta ubicación debido a la mejor circulación sanguínea que se obtiene en esa área. Este enfoque permite una monitorización más confiable de los parámetros cardiovasculares, optimizando la captación de datos esenciales para el control y seguimiento de la condición hipertensiva de la paciente.

## Figura 7

### *Implementación del prototipo*



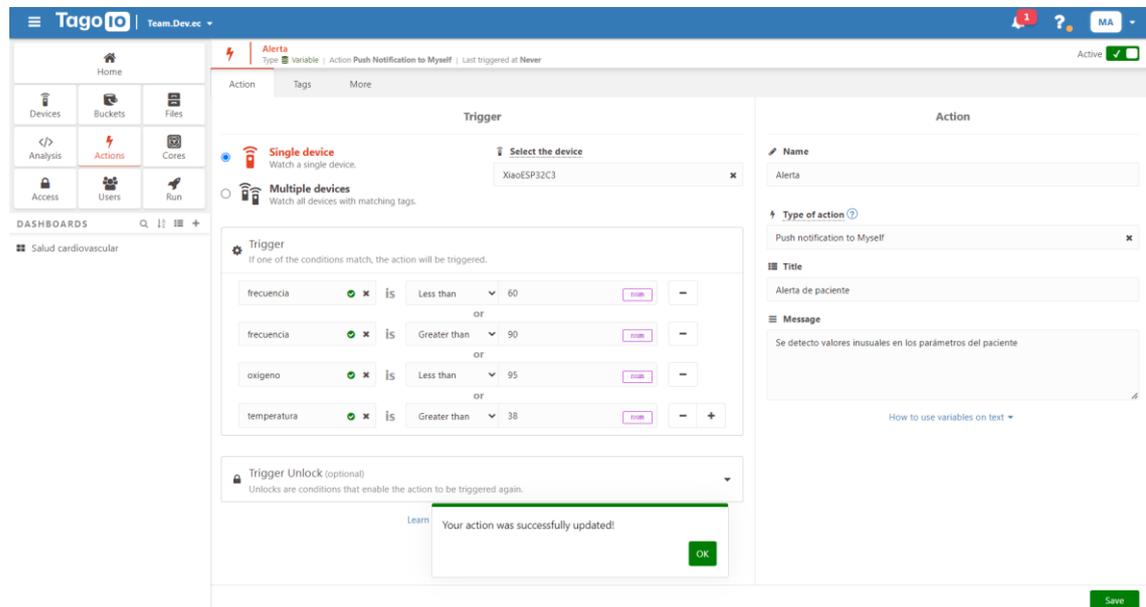
### **Configuración de Alertas en la plataforma TagoIO**

En la figura 8 se muestran las configuraciones diseñadas para alertar al tutor del paciente sobre cualquier alteración en la frecuencia cardíaca, los niveles de oxígeno en la sangre y la temperatura corporal. En cuanto a la frecuencia cardíaca, se consideran normales los valores que oscilan entre 60 y 100 latidos por minuto. Los niveles de oxígeno en la sangre se consideran óptimos cuando se encuentran entre el 95% y el 100%, mientras que la temperatura corporal normal varía entre 36,5°C y 37,5°C.

El sistema de alerta temprana está configurado para enviar notificaciones inmediatas al tutor a través de la plataforma, garantizando así una respuesta rápida ante cualquier desviación de estos parámetros que pueda indicar un problema de salud inminente. Esta funcionalidad es crucial para el monitoreo continuo y proactivo del estado de salud del paciente, proporcionando tranquilidad tanto al paciente como a su tutor.

## Figura 8

### Configuración de alerta temprana en TagoIO



### Aplicación de Modelos de Maching Learning para predicción de alteraciones en las variables de salud cardiovascular

Se integraron varios modelos para el desarrollo de la presente propuesta, el primer modelo fue de correlaciones entre variables (Schober et al., 2018), el cual permite cuantificar relaciones lineales entre diferentes factores. Destaca la importancia de las correlaciones para detectar patrones.

Adicional un modelo de regresión que utiliza el método de permutación (Ciszewski et al., 2024) que permite predecir alteraciones en alguna de las variables consideradas que afecte directamente a la salud cardiovascular del paciente. Además, el modelo también se enfoca en la importancia de cada variable de estudio en la predicción del resultado final.

También se aplicó un modelo LSTM (Rincón-Arango et al., 2020), el cual consiste en aplicar una red neuronal como método de aprendizaje de datos y almacenarlo en un periodo corto de tiempo para predecir alteraciones en las variables de estudio mediante patrones a partir de los datos históricos.

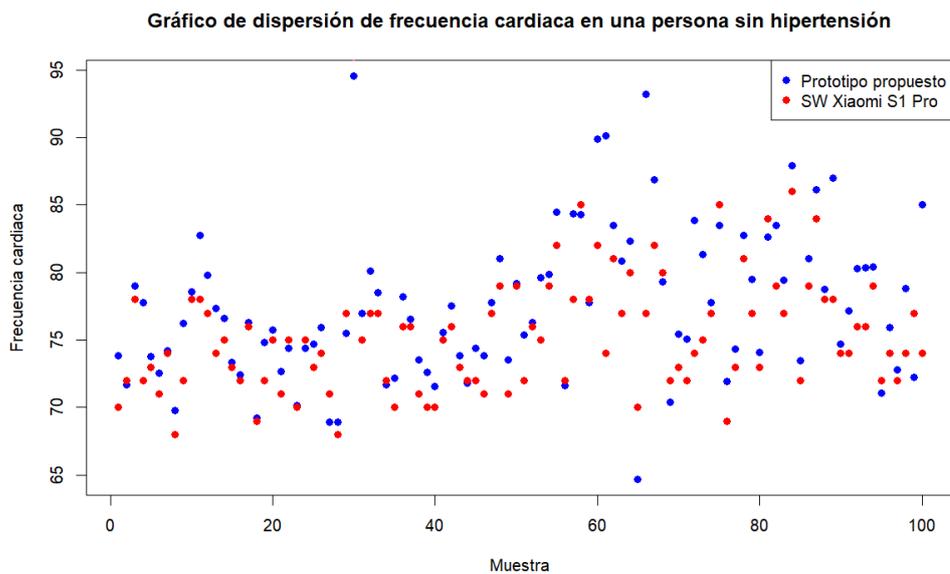
# RESULTADOS

## Análisis de datos de frecuencia cardíaca

Con base en el gráfico de dispersión y los datos proporcionados mostrados en la figura 9, de acuerdo al análisis estadístico comparativo de las lecturas de frecuencia cardíaca entre el prototipo propuesto y el smartwatch Xiaomi S1 Pro implementada en una persona sin hipertensión. La media de las lecturas del prototipo es de 77.4901 latidos por minuto, ligeramente superior a la media del smartwatch, que es de 75.38 latidos por minuto. Esto indica que, en promedio, el prototipo tiende a medir frecuencias cardíacas un poco más altas. La mediana, que representa el valor central de las distribuciones, es de 76.565 para el prototipo y de 75 para el smartwatch, mostrando una tendencia similar. Las modas, o los valores más frecuentemente observados, presentan una mayor variabilidad en el prototipo (68.91, 74.39, 74.67, 77.75, 82.78, 83.47) en comparación con una única moda de 72 en el smartwatch, sugiriendo que las lecturas del prototipo tienen una mayor dispersión o diversidad. En general, aunque las lecturas de ambos dispositivos son comparables y siguen patrones similares, el prototipo parece tener una ligera inclinación a medir frecuencias más altas y presenta una mayor variabilidad en sus lecturas.

### Figura 9

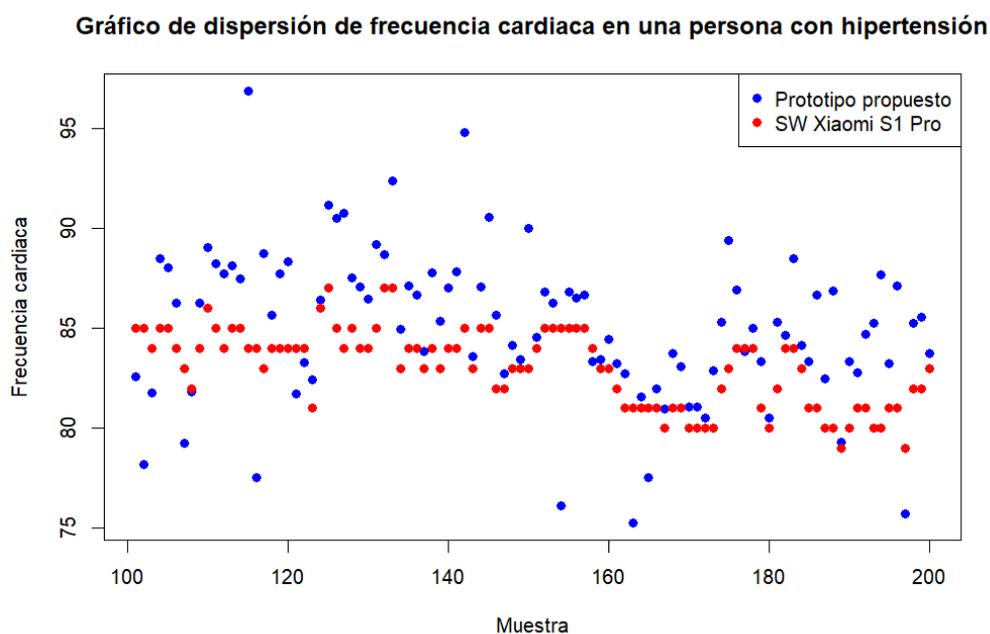
Gráfico de dispersión de FC en una persona sin hipertensión



El gráfico de dispersión presentado en la figura 10, muestra las frecuencias cardíacas de una persona con hipertensión, donde los puntos azules representan los datos obtenidos del prototipo propuesto y los puntos rojos corresponden al smartwatch Xiaomi S1 Pro. Al analizar los datos proporcionados, se observa que la media de las frecuencias cardíacas para el prototipo es de 85.0851, con una mediana de 85.265 y varias modas alrededor de 83.36, 83.84, 85.23, 86.25 y 87.1. En comparación, el smartwatch tiene una media de 83.03, una mediana de 83.5 y una moda única de 84. Visualmente, los puntos rojos del smartwatch parecen más agrupados alrededor de una frecuencia cardíaca constante en comparación con los puntos azules del prototipo, que muestran una mayor variabilidad. Esto sugiere que el smartwatch podría estar proporcionando lecturas más consistentes, mientras que el prototipo tiene una mayor dispersión de datos. La diferencia en las medidas de tendencia central y la variabilidad visualmente observada podrían indicar diferencias en la precisión y consistencia entre los dos dispositivos de medición. Sin embargo, es importante mencionar que la frecuencia cardíaca de una persona oscila entre 60 y 100 latidos por minuto, por lo tanto, los datos proporcionados por el dispositivo se consideran confiables.

### Figura 10

*Gráfico de dispersión de FC en una persona hipertensa*



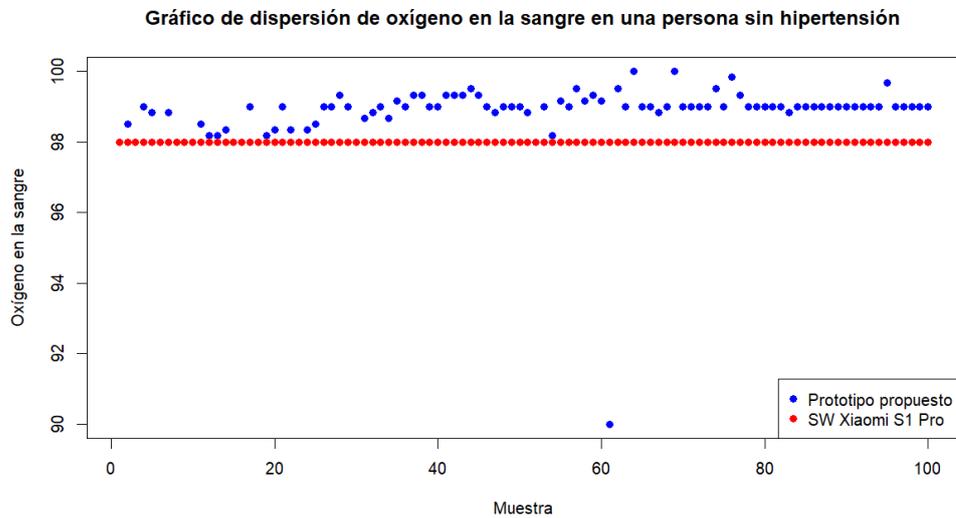
El análisis estadístico comparativo de las lecturas de frecuencia cardíaca entre el prototipo propuesto y el smartwatch Xiaomi S1 Pro latidos por minuto, ligeramente superior a la media del smartwatch de 75.38, y una mediana de 76.565 frente a 75, lo que indica una tendencia a medir frecuencias cardíacas ligeramente más altas. Aunque ambos dispositivos siguen patrones similares, el prototipo presenta una mayor dispersión en sus lecturas, evidenciada por múltiples valores de moda en comparación con la única moda de 72 del smartwatch. Esta variabilidad sugiere una menor consistencia en las mediciones del prototipo. Sin embargo, dado que las medias y medianas de ambos dispositivos son bastante cercanas, se puede concluir que el prototipo es razonablemente confiable, estimándose su confiabilidad en un 90%.

### **Análisis de datos de oxígeno en la sangre**

El gráfico de dispersión presentado en la figura 11 muestra los niveles de oxígeno en la sangre de una persona sin hipertensión, donde los puntos azules representan los datos obtenidos del prototipo propuesto y los puntos rojos corresponden al smartwatch Xiaomi S1 Pro. Según los datos proporcionados, la media del prototipo es de 98.788, con una mediana de 99 y una moda de 99. En contraste, los datos del smartwatch tienen una media de 98, una mediana de 98 y una moda de 98. Visualmente, los puntos rojos del smartwatch están alineados de manera muy constante en el valor de 98, lo que indica poca variabilidad en las mediciones. Por otro lado, los puntos azules del prototipo muestran una mayor dispersión alrededor del valor de 99, sugiriendo que el prototipo presenta una variabilidad mayor en sus lecturas de oxígeno en la sangre. Esta diferencia en la dispersión y las medidas de tendencia central entre los dos dispositivos no implica una medición inconsistente, esta tendencia de diferencias radica principalmente en que el smartwatch presenta valores enteros, mientras que el prototipo muestra valores con punto decimal.

## Figura 11

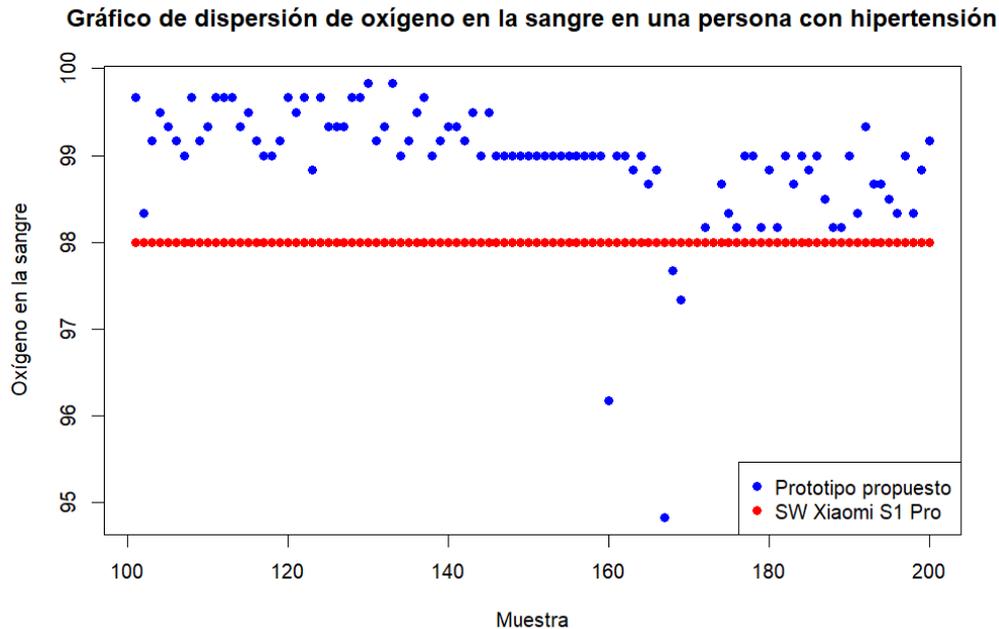
*Gráfico de dispersión de SPO2 en una persona sin hipertensión*



El gráfico de dispersión presentado en la figura 12 muestra los niveles de oxígeno en la sangre de una persona con hipertensión, donde los puntos azules representan los datos obtenidos del prototipo propuesto y los puntos rojos corresponden al smartwatch Xiaomi S1 Pro. Según los datos proporcionados, la media del prototipo es de 98.9203, con una mediana y una moda de 99. Por otro lado, los datos del smartwatch tienen una media, mediana y moda de 98. Visualmente, se observa que los puntos rojos del smartwatch están alineados consistentemente en el valor de 98, indicando poca variabilidad en las mediciones. En contraste, los puntos azules del prototipo muestran una mayor dispersión alrededor del valor de 99, lo que sugiere que el prototipo tiene una mayor variabilidad en sus lecturas de oxígeno en la sangre. Esta diferencia en la dispersión y las medidas de tendencia central entre ambos dispositivos se forma a partir de que el reloj inteligente muestra valores enteros mientras que el prototipo propuesto emite valores con punto decimal para una mejor comprensión de los datos.

**Figura 12**

*Gráfico de dispersión de oxígeno en la sangre en una persona hipertensa*



Ambos dispositivos comparados tienen medias y medianas cercanas (98.788 para el prototipo y 98 para el smartwatch), y la moda del prototipo es 99 frente a 98 del smartwatch, el prototipo exhibe una mayor dispersión de datos en comparación con la consistencia del smartwatch, cuya alineación constante en el valor de 98 sugiere una menor variabilidad. Sin embargo, esta mayor variabilidad del prototipo, que muestra valores con punto decimal, no necesariamente indica inconsistencia en las mediciones. Dado que las diferencias radican principalmente en la precisión decimal del prototipo y que ambas medidas centrales están muy alineadas, se puede estimar la confiabilidad del prototipo en un 90%.

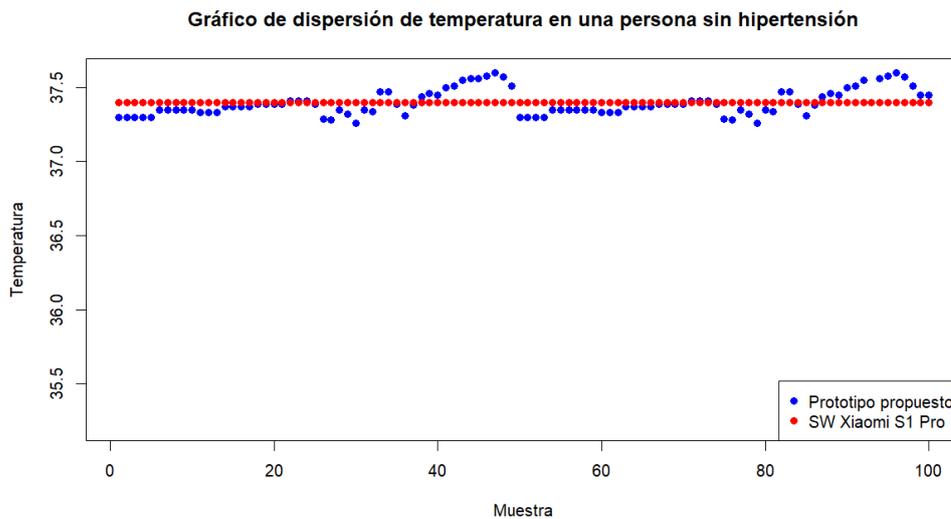
### **Análisis de datos de temperatura corporal**

El gráfico de dispersión presentado en la figura 13 muestra la temperatura de una persona sin hipertensión, comparando las lecturas del prototipo propuesto (en azul) con las del smartwatch Xiaomi S1 Pro (en rojo). Los puntos rojos (Xiaomi S1 Pro) se distribuyen de manera más uniforme alrededor de la línea de 37.4°C, coincidiendo con sus medidas estadísticas: una media, mediana y moda de 37.4°C. Por otro lado, los puntos azules (prototipo propuesto) presentan mayor variabilidad, con temperaturas que oscilan más

ampliamente, aunque su media es  $37.374^{\circ}\text{C}$ , la mediana es  $37.37^{\circ}\text{C}$  y la moda es  $37.35^{\circ}\text{C}$ . Esto sugiere que el prototipo propuesto puede tener una variabilidad ligeramente mayor en sus lecturas de temperatura en comparación con el smartwatch Xiaomi S1 Pro, que muestra una estabilidad más consistente en sus datos.

### Figura 13

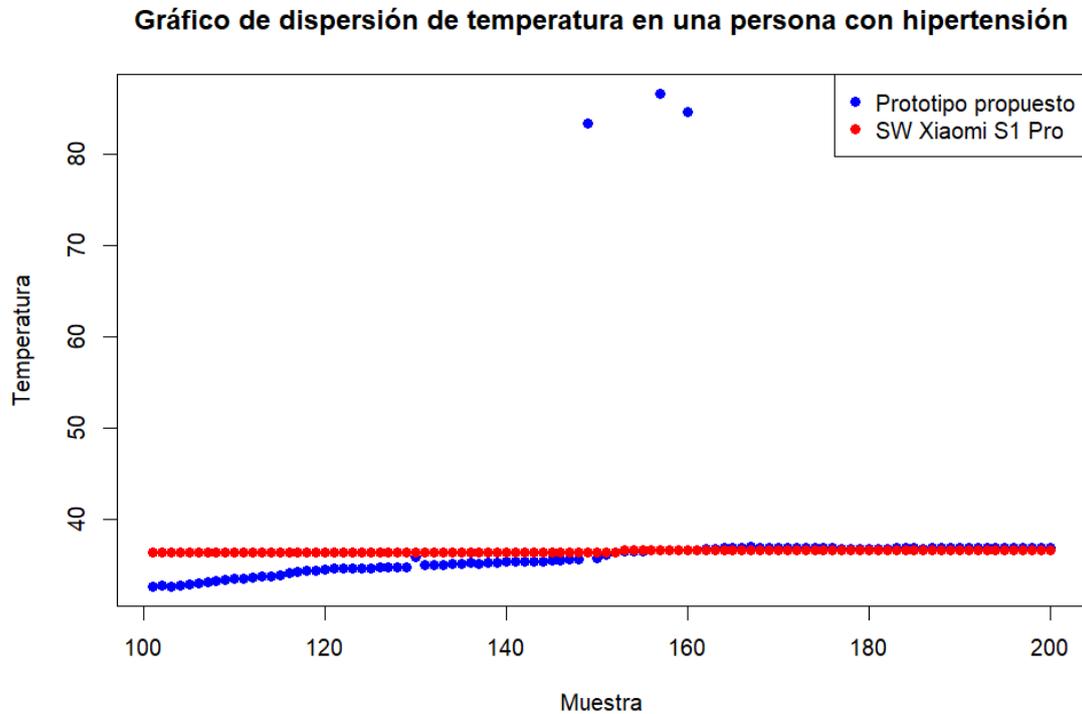
*Gráfico de dispersión de la temperatura en una persona sin hipertensión*



El gráfico de dispersión presentado en la figura 14 muestra la temperatura de una persona con hipertensión, comparando las lecturas del prototipo propuesto (en azul) con las del smartwatch Xiaomi S1 Pro (en rojo). Los puntos rojos del Xiaomi S1 Pro están agrupados en torno a  $36.4^{\circ}\text{C}$ , lo cual concuerda con su media y mediana de  $36.496^{\circ}\text{C}$  y  $36.4^{\circ}\text{C}$  respectivamente, mostrando una distribución uniforme y consistente. En contraste, los puntos azules del prototipo propuesto muestran una mayor variabilidad, con algunos valores significativamente altos, y estadísticas más dispersas: una media de  $37.1055^{\circ}\text{C}$ , una mediana de  $36.29^{\circ}\text{C}$  y múltiples modas ( $6.81$ ,  $36.84$ ,  $36.87$ ). Esto sugiere que el prototipo tiene una mayor dispersión en las mediciones, con varios puntos atípicos, comparado con la estabilidad del Xiaomi S1 Pro.

## Figura 14

Gráfico de dispersión de la temperatura en una persona hipertensa



Ambos dispositivos comparados tienen medias, medianas y modas muy cercanas (el prototipo con una media de  $37.374^{\circ}\text{C}$ , una mediana de  $37.37^{\circ}\text{C}$  y una moda de  $37.35^{\circ}\text{C}$ , y el smartwatch con  $37.4^{\circ}\text{C}$  en todas las medidas), el prototipo presenta una mayor variabilidad en sus lecturas de temperatura. Los puntos rojos del smartwatch Xiaomi S1 Pro se distribuyen uniformemente alrededor de la línea de  $37.4^{\circ}\text{C}$ , indicando una estabilidad más consistente en comparación con los puntos azules del prototipo, que oscilan más ampliamente. Esta mayor variabilidad sugiere que el prototipo tiene una precisión ligeramente menor. No obstante, debido a la proximidad de las medidas centrales, se puede estimar la confiabilidad del prototipo en un 92%.

### Autonomía del dispositivo

A continuación, en la tabla 7 se presenta el consumo individual de la unidad de control ESP32C3 y los sensores integrados al dispositivo electrónico. Además, se presenta la suma total del consumo en unidad de miliamperios.

**Tabla 7***Consumo total de corriente del prototipo propuesto*

Descripción	Consumo de corriente
Xiao ESP32C3	75 [mA]
MLX90614	0.17 [mA]
MAX30100	1.2 [mA]
Consumo total	76.37 [mA]

De acuerdo a las características de la batería de litio, las cuales son de 3.7 [V] y 500 mAh, la autonomía del dispositivo en horas es:

$$\textit{Autonomía} = \frac{\textit{Capacidad de la batería}}{\textit{Consumo del dispositivo}}$$

$$\textit{Autonomía} = \frac{500 \text{ [mAh]}}{76.37 \text{ [mA]}}$$

$$\textit{Autonomía} = \frac{500 \text{ [mAh]}}{76.37 \text{ [mA]}}$$

$$\textit{Autonomía} \approx 6,54 \text{ [h]}$$

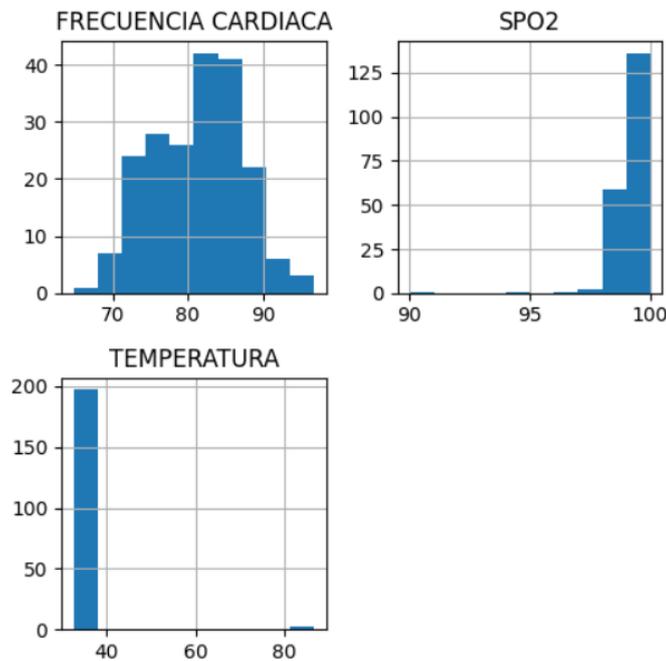
### **Interpretación de histogramas con 200 datos**

Para los histogramas mostrados en la figura 15, se tomó una muestra de 200 datos donde se obtuvieron los siguientes resultados: la distribución de la frecuencia cardíaca muestra una distribución aproximadamente normal. Los valores oscilan entre 70 y 95 latidos por minuto, con la mayoría de las observaciones concentradas alrededor de 80 a 90 bpm. Esto sugiere que la mayoría de los individuos tienen una frecuencia cardíaca en ese rango, que puede ser considerado normal para adultos en reposo. La distribución de la SpO2 muestra una asimetría hacia la derecha. La mayoría de los valores están concentrados cerca del 100%, con muy pocas observaciones por debajo del 95%. Esto indica que la mayoría de los individuos tienen niveles de oxígeno en la sangre que son considerados normales y saludables. La distribución de la temperatura muestra una gran concentración de valores

en el rango de los 35-40 grados Celsius. Existen algunos valores atípicos que llegan hasta cerca de 80, lo cual es inusual y podría indicar errores de medición o datos anómalos. La mayoría de las temperaturas corporales normales están entre 36.5 y 37.5 grados Celsius, por lo que la gran concentración en este rango es esperada.

### Figura 15

*Histograma de variables con una muestra de 200 datos*



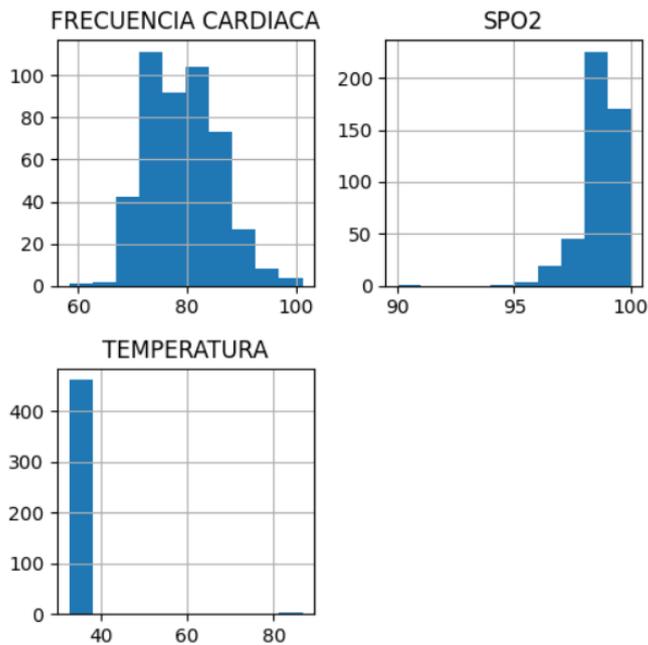
### Interpretación de histogramas con 464 datos

Para los histogramas mostrados en la figura 16, se tomó una muestra de 464 datos donde se obtuvieron los siguientes resultados: la distribución de la El histograma muestra una distribución aproximadamente normal centrada entre 75 y 80 latidos por minuto. La mayoría de los valores oscilan aproximadamente entre 60 y 100 bpm, con un pico de alrededor de 70 a 80 bpm. Hay pocos valores atípicos por debajo de 60 y por encima de 100 bpm. La distribución está sesgada a la derecha, con la mayoría de los valores agrupados entre el 95% y el 100%. Un número significativo de mediciones están en o muy cerca del 100%. Muy pocas lecturas caen por debajo del 90%, lo que indica que la mayoría de los sujetos tienen niveles de saturación de oxígeno normales o altos. El histograma de temperatura muestra un rango muy estrecho con un pico pronunciado alrededor de 40°C. Esta distribución inusual podría indicar un problema con los datos o

el sensor, ya que la temperatura del cuerpo humano normalmente oscila entre 36°C y 37,5°C en condiciones normales. La presencia de valores atípicos lejos del pico sugiere que podría haber lecturas o entradas erróneas.

### Figura 16

*Histograma de variable con una muestra de 464 datos*



### Modelo Maching Learning: correlación entre variables

#### Entrenamiento 1

A continuación, en la figura 17 se presenta la matriz de correlación entre variables con el siguiente análisis estadístico:

#### Correlación entre FH y SpO2: 0.037

La correlación es positiva pero muy cercana a 0, lo que indica que no hay una relación significativa entre la frecuencia cardíaca y el nivel de oxígeno en la sangre. Es decir, los cambios en la frecuencia cardíaca no están asociados con el nivel de oxígeno en la sangre.

#### Correlación entre FH y Temp: -0.038

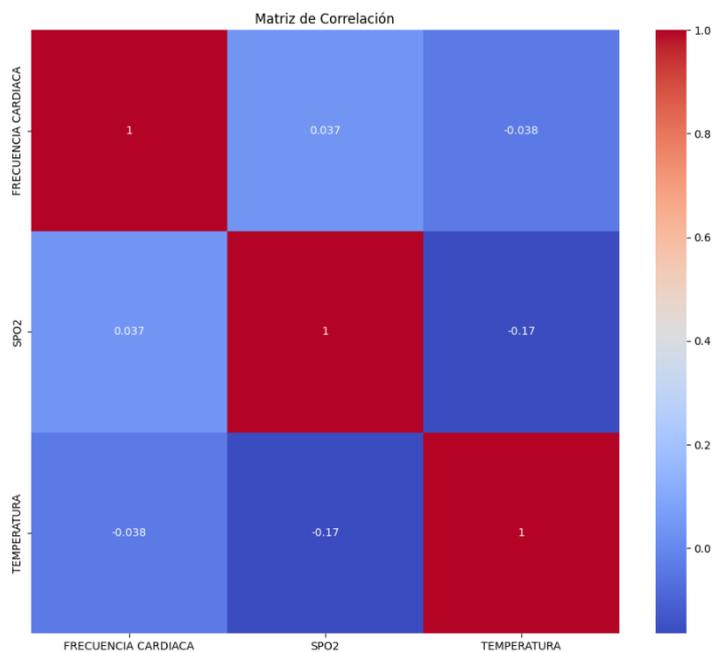
La correlación es negativa pero muy cercana a 0, lo que también sugiere que no hay una relación significativa entre la frecuencia cardíaca y la temperatura. Los cambios en la frecuencia cardíaca no están asociados con cambios en la temperatura corporal.

### **Correlación entre SpO2 y Temp: -0.17**

La correlación es negativa y un poco más fuerte que las anteriores, aunque sigue siendo relativamente débil. Esto sugiere que hay una ligera tendencia a que cuando el nivel de oxígeno en la sangre disminuye, la temperatura tiende a aumentar y viceversa, aunque esta relación no es muy fuerte.

### **Figura 17**

*Gráficos la matriz de correlación en una muestra de 200 datos*



### **Entrenamiento 2**

A continuación, en la figura 18 se presenta la matriz de correlación entre variables con el siguiente análisis estadístico:

### **Correlación entre la Frecuencia Cardíaca y SpO2: 0.14**

Este valor indica una correlación positiva muy baja. En otras palabras, cuando una de estas variables aumenta, la otra tiende a aumentar ligeramente, pero la relación es débil.

### **Correlación entre la Frecuencia Cardíaca y Temperatura: -0.045.**

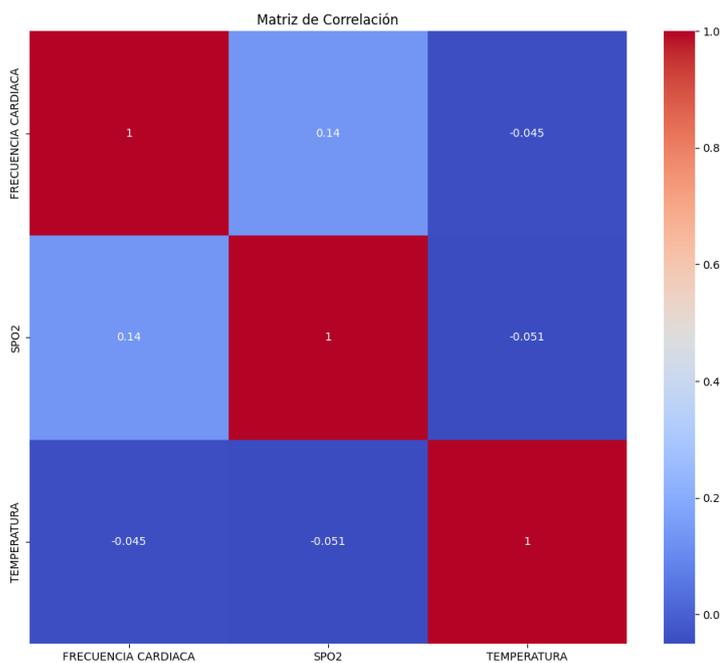
Este valor indica una correlación negativa muy baja. Esto significa que cuando una de estas variables aumenta, la otra tiende a disminuir ligeramente, pero esta relación también es muy débil.

### **Correlación entre SpO2 y Temperatura: -0.051**

Este valor indica una correlación negativa muy baja. Esto significa que cuando una de estas variables aumenta, la otra tiende a disminuir ligeramente, pero al igual que las otras, esta relación es muy débil.

### **Figura 18**

*Gráficos la matriz de correlación en una muestra de 464 datos*



### **Modelo Maching Learning: Modelo de regresión utilizando método de permutación**

#### **Entrenamiento 1**

A continuación, en la figura 19 se presenta el gráfico estadístico de permutación con una muestra de 200 datos, donde las características son la temperatura y el oxígeno en la sangre para predecir la frecuencia cardíaca en un paciente:

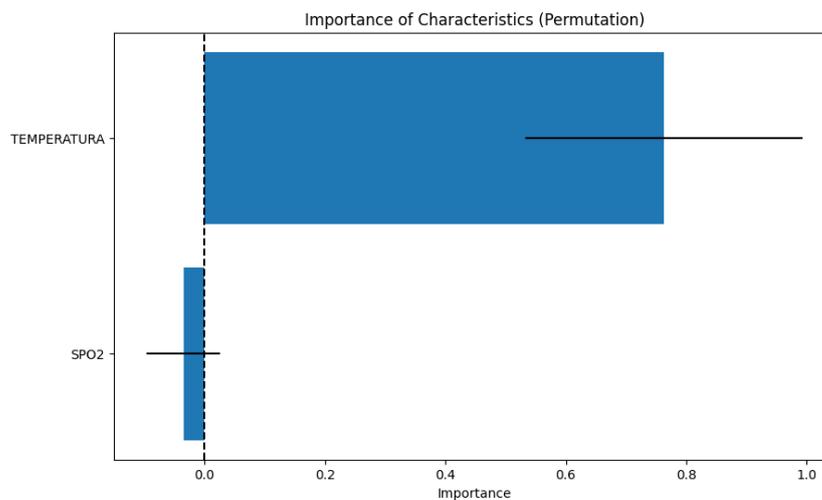
### Importancia de las Características:

La temperatura tiene una importancia mucho mayor en el modelo, con un valor de aproximadamente 0.75. Esto indica que la temperatura es una característica muy significativa para predecir la frecuencia cardiaca con el modelo propuesto. Mientras que el oxígeno en la sangre (SpO2) tiene una importancia significativamente menor, con un valor alrededor de 0.15. Aunque contribuye a la predicción, su impacto es mucho menor comparado con la temperatura.

La variable TEMPERATURA no empieza desde 0 y SPO2 empieza desde un valor negativo porque la importancia de la característica calculada por el método de permutación puede dar valores negativos, lo que indica que permutar esta característica resultó en una mejor predicción, lo cual es un indicador de ruido o irrelevancia de la característica para el modelo.

### Figura 19

*Gráfico de permutación en una muestra de 200 datos*



### Entrenamiento 2

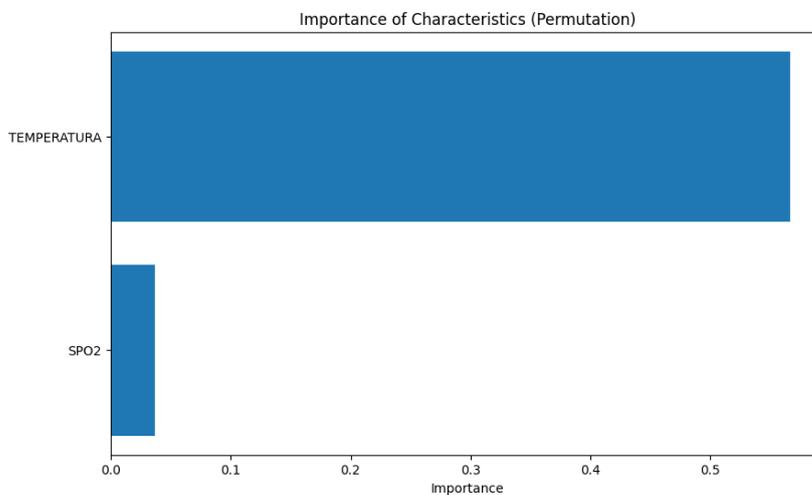
A continuación, en la figura 20 se presenta el gráfico estadístico de permutación en una muestra de 464 datos, donde las características son la temperatura y el oxígeno en la sangre para predecir la frecuencia cardiaca en un paciente:

### Importancia de las Características:

La barra correspondiente a la "TEMPERATURA" tiene una importancia significativamente mayor que la de "SPO2". Esto indica que, en el contexto de tu modelo de regresión, la temperatura es una variable mucho más importante para predecir la frecuencia cardíaca. Aunque "SPO2" también muestra algo de importancia, es considerablemente menor en comparación con la temperatura. Esto sugiere que, aunque la SpO2 puede tener algún efecto sobre la frecuencia cardíaca, su impacto no es tan fuerte como el de la temperatura en tu modelo.

### Figura 20

Gráfico de permutación en una muestra de 464 datos



### Modelo Maching Learning: Modelo LSTM de redes neuronales

#### Entrenamiento 1

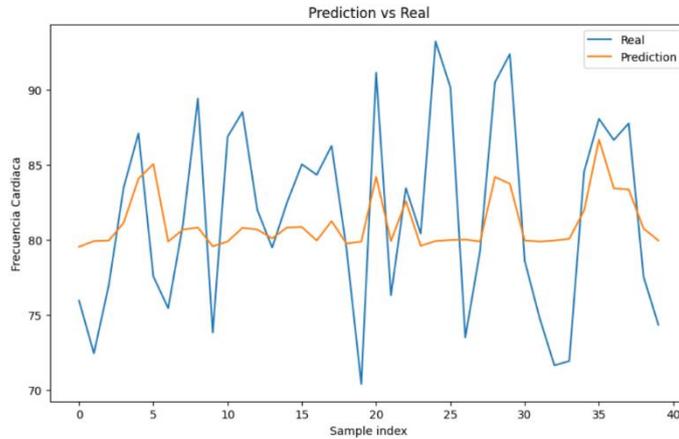
A continuación, en la figura 21, se presenta el gráfico del comportamiento de la frecuencia cardíaca (línea azul) y la predicción del modelo usando redes neuronales LSTM (Long Short Term Memory) (línea naranja) con una muestra de 200 datos.

Se observa que, en algunas secciones, las predicciones capturan bien los picos y valles de los valores reales, mientras que, en otras secciones, hay un desfase significativo. Las áreas donde el modelo no predice bien pueden indicar que el modelo no ha capturado completamente los datos. Esto se debe a la cantidad de datos para entrenar el modelo. Aunque las predicciones no son perfectas, si el modelo captura las tendencias generales de la frecuencia cardíaca, podría ser útil para aplicaciones prácticas donde pequeñas desviaciones no sean críticas.

## Figura 21

Gráfico obtenido del modelo LSTM con una muestra de 200 datos

RMSE: 5.667651383965889  
MAE: 4.721982725219726



## Entrenamiento 2

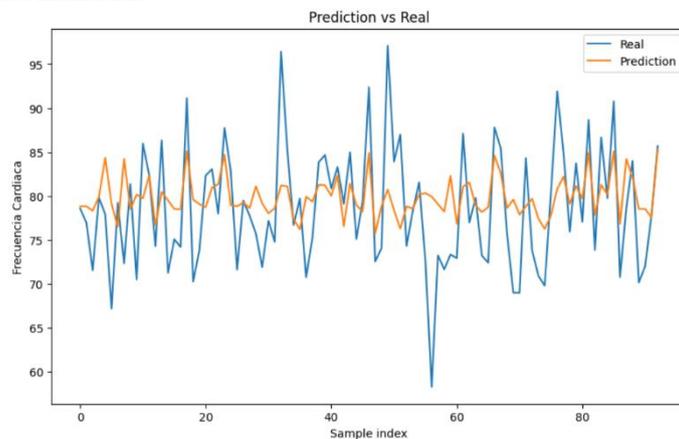
A continuación, en la figura 22, se presenta el gráfico del comportamiento de la frecuencia cardiaca (línea azul) y la predicción del modelo usando redes neuronales LSTM (Long Short Term Memory) (línea naranja) con una muestra de 464 datos.

Las predicciones parecen suavizar las fluctuaciones presentes en los datos reales. Es decir, los picos altos y valles profundos en los valores reales no se reflejan con la misma intensidad en las predicciones. Las predicciones tienden a mantenerse más estables en comparación con las variaciones bruscas observadas en los datos reales.

## Figura 22

Gráfico obtenido del modelo LSTM con una muestra de 464 datos

RMSE: 6.206451608722289  
MAE: 4.948854639196909



### **Análisis de RMSE (Raíz del error cuadrático medio)**

El error cuadrático medio de las predicciones es de aproximadamente 6.206 unidades. Este valor da una idea de la magnitud de los errores de predicción, y al penalizar más los errores grandes, puede destacar la presencia de grandes discrepancias en algunas predicciones.

### **Análisis de MAE (Error absoluto medio)**

MAE: 4.949: Indica que, en promedio, el error absoluto medio de las predicciones es de aproximadamente 4.949 unidades. Este valor da una medida más directa y menos sensible a los errores grandes en comparación con el RMSE.

## CONCLUSIONES

- Se logró desarrollar un dispositivo electrónico para la medición de variables relevantes para la salud cardiovascular. Este dispositivo puede monitorear parámetros como la frecuencia cardíaca, la presión arterial y la oxigenación sanguínea con precisión, demostrando ser una herramienta viable para el monitoreo domiciliario.
- El modelo de regresión con método de permutación se puede aplicar en un monitoreo de pacientes, porque prioriza el seguimiento de la temperatura como un indicador crítico sobre alteraciones en la frecuencia cardíaca.
- De acuerdo a los datos, el modelo de predicción mediante redes neuronales LSTM presenta 4,949 de error absoluto en la predicción de los datos. Sin embargo, el modelo se considera viable y se puede emplear en el dispositivo de forma continua.
- En cuanto a la autonomía, el dispositivo propuesto incorpora una batería de 500 [mAh] que permite un funcionamiento continuo de aproximadamente 7 horas. La autonomía puede incrementar debido a que el valor del consumo seleccionado de la unidad de control es cuando realiza el envío de datos a la plataforma IoT, mientras no realice esa acción el consumo baja a  $<25$  [mA].
- La carcasa del dispositivo, realizada en impresión 3D, está optimizada para adaptarse perfectamente a los componentes electrónicos. En cuanto a la movilidad del paciente, el uso del dispositivo no causa molestias. Sin embargo, se han observado picos en los valores medidos cuando el paciente realiza movimientos bruscos.

## **RECOMENDACIONES**

- Realizar mantenimientos preventivos y correctivos al dispositivo propuesto, de manera que se eviten valores excesivamente dispersos en cada variable monitoreada por efectos de suciedad o calibración de sensores.
- Para el correcto funcionamiento del dispositivo propuesto, es necesario que esté conectado a una red WiFi con acceso a internet, de tal manera que garantice así el envío correcto de los datos recolectados a la plataforma IoT TagoIO.
- Aunque el dispositivo no causa molestias durante su uso, se recomienda programar intervalos de descanso para evitar que el paciente se sienta estresado por el monitoreo continuo de su frecuencia cardíaca, nivel de oxígeno en la sangre o temperatura.

## REFERENCIAS

- Aditya, L. (2020). Rancang bangun alat pengukur kadar oksigen non invasive menggunakan sensor max30100. *Jurnal Ilmiah Elektrokrisna*, 8(2), 62-69.
- Barrenech, Ó. (2023, mayo 16). *Informe de Ecuador: Mejorando la salud cardiovascular desde comunidades locales hasta el nivel nacional con un enfoque participativo - OPS/OMS / Organización Panamericana de la Salud*. <https://www.paho.org/es/noticias/16-5-2023-informe-ecuador-mejorando-salud-cardiovascular-desde-comunidades-locales-hasta>
- Buitrón, D. (2024). *Arquitecturas y modelos de referencia para sistemas IoT: estado del arte de las arquitecturas para sistemas IoT*. [bachelorThesis, Escuela Politécnica Nacional]. <http://bibdigital.epn.edu.ec/handle/15000/22368>
- Ciszewski, M., Söhl, J., Leenen, T., van Trigt, B., & Jongbloed, G. (2024). *Testing for no effect in regression problems: A permutation approach* (arXiv:2305.02685). arXiv. <https://doi.org/10.48550/arXiv.2305.02685>
- Espressif Systems. (2024). *ESP32-C3 Series* (Datasheet 1.2; ESP32-C3, p. 45). Espressif Systems. [https://files.seeedstudio.com/wiki/Seeed-Studio-XIAO-ESP32/esp32-c3\\_datasheet.pdf](https://files.seeedstudio.com/wiki/Seeed-Studio-XIAO-ESP32/esp32-c3_datasheet.pdf)
- Forum Arduino. (2022). *Esp8266 con Max30100 y MLX90614* [Foro de soporte de Arduino]. Esp8266 con Max30100 y MLX90614. <https://forum.arduino.cc/t/esp8266-con-max30100-y-mlx90614/981189>
- Fritzing. (s. f.). *Download Fritzing*. Recuperado 29 de mayo de 2024, de <https://fritzing.org/>

- Garces, A., Manzano, S., Nuñez, C., Pallo, J., & Jurado, M. (2021). Plataforma IoT de bajo-coste para aplicaciones de telemedicina. *Universidad Técnica de Ambato*, 43, 153-166.
- Manzano, E., & Bejarano, M. (2021). Implementación de un Sistema de Monitoreo a nivel de prototipo de signos vitales. *Universidad Nacional de Trujillo*, 14, 17-40. <https://doi.org/10.26439/interfases2021.n014.5168>
- Maxim Integrated. (2014). *Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health* (Datasheet 0; MAX30100, p. 29). Analog Devices. [https://components101.com/sites/default/files/component\\_datasheet/MAX30100-Datasheet.pdf](https://components101.com/sites/default/files/component_datasheet/MAX30100-Datasheet.pdf)
- Melexis Inspired Engineering. (2019). *MLX90614 family* (Datasheet 13; MLX90614, p. 57). Melexis inspired Engineering. <https://media.melexis.com/-/media/files/documents/datasheets/mlx90614-datasheet-melexis.pdf>
- Naranjo, N. (2016). Teleconsulta: Análisis de la herramienta de consulta médica virtual en el sistema de salud pública. *Journal of Science and Research: Revista Ciencia e Investigación*, 1(2), 23-26. <https://doi.org/10.26910/issn.2528-8083vol1iss2.2016pp23-26>
- OPS. (2019). *La Carga de Enfermedades Cardiovasculares—OPS/OMS | Organización Panamericana de la Salud*. <https://www.paho.org/es/enlace/carga-enfermedades-cardiovasculares>
- Rincón-Arango, J. A., Guerra-Ojeda, S., Carrascosa Casamayor, C., & Julian, V. (2020). IoT and fog computing-based monitoring system for cardiovascular

- patients with automatic ECG classification using deep neural networks. *Sensors*, 20(24), 1-19. <https://doi.org/10.3390/s20247353>
- Schober, P., Boer, C., & Schwarte, L. A. (2018). Correlation Coefficients: Appropriate Use and Interpretation. *Anesthesia & Analgesia*, 126(5), 1763. <https://doi.org/10.1213/ANE.0000000000002864>
- TagoIO. (2023). *TagoIO: IoT platform for data management and real-time analysis* [Sitio web]. <https://www.tago.io>
- Vela, J., Mera, E., & Palacios, E. (2024). Propuesta de un sistema de monitoreo de indicadores de salud mediante IOT para la toma de signos vitales. *Revista Social Fronteriza*, 4(1), 14. [https://doi.org/10.59814/resofro.2024.4\(1\)169](https://doi.org/10.59814/resofro.2024.4(1)169)
- Witt, G. F. V., Ronquillo, M. C. J., Bravo, E. M. G., & Rodríguez, P. de las M. W. (2023). Desafíos y problemas de la salud pública en Ecuador. *RECIAMUC*, 7(2), Article 2. [https://doi.org/10.26820/reciamuc/7.\(2\).abril.2023.10-21](https://doi.org/10.26820/reciamuc/7.(2).abril.2023.10-21)
- Xiaomi. (2024). *Xiaomi Watch S1 Pro Smart Watch User Manual* (Manual 1; S1 PRO, p. 8). Xiaomi Communications Co.m Ltd. <https://manuals.plus/xiaomi/watch-s1-pro-smart-watch-manual.pdf>
- Montaño, M., Santos, V., Jaramillo, D., Torres Guin, W., & Chuquimarca Jiménez, L. (2023). *Empowering Low-Power Wide-Area Networks: Unlocking the Potential of Sigfox in Local Transmission*. 417-429. [https://doi.org/10.1007/978-3-031-48930-3\\_32](https://doi.org/10.1007/978-3-031-48930-3_32)
- Montaño-Blacio, M., González-Escarabay, J., Jiménez-Sarango, Ó., Mingo-Morocho, L., & Carrión-Aguirre, C. (2023). Diseño y despliegue de un sistema de

monitoreo basado en IoT para cultivos hidropónicos. *Ingenius*, 30, 9-18.

<https://doi.org/10.17163/ings.n30.2023.01>

# ANEXOS

## Anexo 1

*Código en Google Colab de histograma y correlación con 200 datos de muestra*

```
import pandas as pd
# Cargar los datos
file_path = '/Datos_dispositivo_tesis.xlsx'
data = pd.read_excel(file_path)
data_master = data.head(200)
# Mostrar las primeras filas del dataframe
print(data_master.head())
# Obtener información general sobre los datos
print(data_master.info())
# Obtener estadísticas descriptivas
print(data_master.describe())

#####
import matplotlib.pyplot as plt
import seaborn as sns
# Distribución de las variables numéricas
data_master.hist(figsize=(5, 5))
plt.tight_layout()
plt.show()
# Visualización de las correlaciones entre las variables
corr_matrix = data_master.corr()
plt.figure(figsize=(12, 10))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Matriz de Correlación')
plt.show()
```

## Anexo 2

*Código en google colab de histograma y correlación con 464 datos de muestra*

```
import pandas as pd
    Cargar los datos
file_path = '/Datos_dispositivo_tesis.xlsx'
data_master = pd.read_excel(file_path)
# Mostrar las primeras filas del dataframe
print(data_master.head())
# Obtener información general sobre los datos
print(data_master.info())
# Obtener estadísticas descriptivas
print(data_master.describe())

#####
import matplotlib.pyplot as plt
import seaborn as sns
# Distribución de las variables numéricas
data_master.hist(figsize=(5, 5))
plt.tight_layout()
plt.show()
# Visualización de las correlaciones entre las variables
corr_matrix = data_master.corr()
plt.figure(figsize=(12, 10))
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Matriz de Correlación')
plt.show()
```

### Anexo 3

*Modelo de regresión utilizando el método de permutación con 200 datos*

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.inspection import permutation_importance
# Seleccionar las características y la variable objetivo
features = data_master[['TEMPERATURA', 'SPO2']]
target = data_master['FRECUENCIA CARDIACA']
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2,
random_state=42)
# Entrenar un modelo de regresión (por ejemplo, RandomForestRegressor)
model = RandomForestRegressor(n_estimators=200, random_state=42)
model.fit(X_train, y_train)
# Calcular la importancia de las características utilizando el método de permutación
perm_importance = permutation_importance(model, X_test, y_test, n_repeats=30,
random_state=42)
# Obtener los resultados de importancia de las características
sorted_idx = perm_importance.importances_mean.argsort()
# Graficar la importancia de las características
plt.figure(figsize=(10, 6))
plt.barh(features.columns[sorted_idx],
perm_importance.importances_mean[sorted_idx],
xerr=perm_importance.importances_std[sorted_idx])
plt.axvline(x=0, color='k', linestyle='--') # Línea vertical en x=0
plt.xlabel('Importance')
plt.title('Importance of Characteristics (Permutation)')
plt.savefig('permutacion.png')
plt.show()
```

## Anexo 4

### *Modelo de regresión utilizando el método de permutación con 646 datos*

```
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.inspection import permutation_importance
# Seleccionar las características y la variable objetivo
features = data_master[['TEMPERATURA', 'SPO2']]
target = data_master['FRECUENCIA CARDIACA']
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2,
random_state=42)
# Entrenar un modelo de regresión (por ejemplo, RandomForestRegressor)
model = RandomForestRegressor(n_estimators=100, random_state=42)
model.fit(X_train, y_train)
# Calcular la importancia de las características utilizando el método de permutación
perm_importance = permutation_importance(model, X_test, y_test, n_repeats=30,
random_state=42)
# Obtener los resultados de importancia de las características
sorted_idx = perm_importance.importances_mean.argsort()
# Graficar la importancia de las características
plt.figure(figsize=(10, 6))
plt.barh(features.columns[sorted_idx],
perm_importance.importances_mean[sorted_idx])
plt.xlabel('Importance')
plt.title('Importance of Characteristics (Permutation)')
plt.savefig('permutacion.png')
plt.show()
```

## Anexo 5

### *Modelo LSTM con 200 datos*

```
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from keras.models import Sequential
from keras.layers import LSTM, Dense
import tensorflow as tf
df = pd.read_csv('/content/base_sin_nan.csv')
# Seleccionar las variables predictoras y la variable objetivo
variables_predictoras = ['Temp', 'SpO2']
variable_objetivo = ['FH']
# Dividir los datos en variables predictoras y objetivo
X = df[variables_predictoras].values
y = df[variable_objetivo].values
# Escalar las variables predictoras
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
random_state=42)
# Reshape de los datos para que sean compatibles con la entrada de la red LSTM
X_train = X_train.reshape((X_train.shape[0], 1, X_train.shape[1]))
X_test = X_test.reshape((X_test.shape[0], 1, X_test.shape[1]))
# Definir la arquitectura del modelo LSTM
model = Sequential()
model.add(LSTM(100, input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(Dense(1))
# Compilar el modelo
model.compile(loss='mean_squared_error', optimizer=
tf.keras.optimizers.SGD(learning_rate=0.01)
)
```

```

# Entrenar el modelo
history = model.fit(X_train, y_train, epochs=200, batch_size=32,
validation_data=(X_test, y_test), verbose=2)

#####3
from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
# Realizar predicciones en los datos de prueba
y_pred = model.predict(X_test)
# Calcular métricas de evaluación
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
mae = mean_absolute_error(y_test, y_pred)
print("RMSE:", rmse)
print("MAE:", mae)
# Graficar los valores reales vs predichos
plt.figure(figsize=(10, 6))
plt.plot(y_test, label='Real')
plt.plot(y_pred, label='Prediction')
plt.title('Prediction vs Real')
plt.xlabel('Sample index')
plt.ylabel('Frecuencia Cardiaca')
plt.legend()
plt.show()

#####
from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt

# Limitar el número de muestras a 500
y_test_500 = y_test[:50]
y_pred_500 = y_pred[:50]

```

```

# Calcular métricas de evaluación para las 500 muestras
rmse = np.sqrt(mean_squared_error(y_test_500, y_pred_500))
mae = mean_absolute_error(y_test_500, y_pred_500)
print("RMSE:", rmse)
print("MAE:", mae)
# Graficar las 500 muestras de los valores reales vs predichos
plt.figure(figsize=(10, 6))
plt.plot(y_test_500, label='Real')
plt.plot(y_pred_500, label='Prediction')
plt.title('Prediction vs Real')
plt.xlabel('Sample index')
plt.ylabel('Frecuencia cardiaca')
plt.legend()
plt.savefig('prediccion.png')
plt.show()

```

## Anexo 6

### *Modelo LSTM con 646 datos*

```

import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from keras.models import Sequential
from keras.layers import LSTM, Dense
import tensorflow as tf
df = pd.read_csv('/content/base_sin_nan.csv')
# Seleccionar las variables predictoras y la variable objetivo
variables_predictoras = ['TEMPERATURA', 'SPO2']
variable_objetivo = ['FRECUENCIA CARDIACA']

```

```

# Dividir los datos en variables predictoras y objetivo
X = df[variables_predictoras].values
y = df[variable_objetivo].values
# Escalar las variables predictoras
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
random_state=42)
# Reshape de los datos para que sean compatibles con la entrada de la red LSTM
X_train = X_train.reshape((X_train.shape[0], 1, X_train.shape[1]))
X_test = X_test.reshape((X_test.shape[0], 1, X_test.shape[1]))
# Definir la arquitectura del modelo LSTM
model = Sequential()
model.add(LSTM(100, input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(Dense(1))
# Compilar el modelo
model.compile(loss='mean_squared_error', optimizer=
tf.keras.optimizers.SGD(learning_rate=0.01)
)
# Entrenar el modelo
history = model.fit(X_train, y_train, epochs=463, batch_size=32,
validation_data=(X_test, y_test), verbose=2)

#####
from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
# Realizar predicciones en los datos de prueba
y_pred = model.predict(X_test)
# Calcular métricas de evaluación
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
mae = mean_absolute_error(y_test, y_pred)

```

```

from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
# Limitar el número de muestras a 464
y_test_500 = y_test[:464]
y_pred_500 = y_pred[:464]
# Calcular métricas de evaluación para las 500 muestras
rmse = np.sqrt(mean_squared_error(y_test_500, y_pred_500))
mae = mean_absolute_error(y_test_500, y_pred_500)
print("RMSE:", rmse)
print("MAE:", mae)
# Graficar las 463 muestras de los valores reales vs predichos
plt.figure(figsize=(10, 6))
plt.plot(y_test_500, label='Real')
plt.plot(y_pred_500, label='Prediction')
plt.title('Prediction vs Real')
plt.xlabel('Sample index')
plt.ylabel('Frecuencia cardiaca')
plt.legend()
plt.savefig('prediccion.png')
plt.show()

#####
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from keras.models import Sequential
from keras.layers import LSTM, Dense
import tensorflow as tf
df = pd.read_csv('/content/base_sin_nan.csv')
# Seleccionar las variables predictoras y la variable objetivo
variables_predictoras = ['TEMPERATURA']

```

```

variable_objetivo = ['FRECUENCIA CARDIACA']
# Dividir los datos en variables predictoras y objetivo
X = df[variables_predictoras].values
y = df[variable_objetivo].values
# Escalar las variables predictoras
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.2,
random_state=42)
# Reshape de los datos para que sean compatibles con la entrada de la red LSTM
X_train = X_train.reshape((X_train.shape[0], 1, X_train.shape[1]))
X_test = X_test.reshape((X_test.shape[0], 1, X_test.shape[1]))
# Definir la arquitectura del modelo LSTM
model = Sequential()
model.add(LSTM(100, input_shape=(X_train.shape[1], X_train.shape[2])))
model.add(Dense(1))
# Compilar el modelo
model.compile(loss='mean_squared_error', optimizer=
tf.keras.optimizers.SGD(learning_rate=0.01)
)
# Entrenar el modelo
history = model.fit(X_train, y_train, epochs=463, batch_size=32,
validation_data=(X_test, y_test), verbose=2)

#####
from sklearn.metrics import mean_squared_error, mean_absolute_error
import matplotlib.pyplot as plt
# Realizar predicciones en los datos de prueba
y_pred = model.predict(X_test)

```

```
# Calcular métricas de evaluación
rmse = np.sqrt(mean_squared_error(y_test, y_pred))
mae = mean_absolute_error(y_test, y_pred)
print("RMSE:", rmse)
print("MAE:", mae)
# Graficar los valores reales vs predichos
plt.figure(figsize=(10, 6))
plt.plot(y_test, label='Real')
plt.plot(y_pred, label='Prediction')
plt.title('Prediction vs Real')
plt.xlabel('Sample index')
plt.ylabel('Frecuencia Cardiaca')
plt.legend()
plt.show()
```

## Anexo 7

*Código de dispositivo desarrollado en la IDE Arduino*

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <Wire.h>
#include "Credential_secrets.h"
#include "Envio_datos_IoT.h"

#define statusWifi D7
bool ledWifi = false;

#include "MAX30100_PulseOximeter.h"
#define REPORTING_PERIOD_MS 2000
PulseOximeter pox;
uint32_t tsLastReport = 0;
uint8_t max30100_address = 0x57;

#include <Adafruit_MLX90614.h>
Adafruit_MLX90614 mlx = Adafruit_MLX90614();
uint8_t irmlx90614_address = 0x5A;

float frecCardiaca[10];
float Sop2[10];
float temp[10];

int count = 0;
float valueHR = 0.0;
float valueOxi = 0.0;
float valueTemp = 0.0;
float valueFrecAux = 0.0;
float valueOxiAux = 0.0;
```

```

float frecMayor = 0.0;
void setup(){
  Serial.begin(115200);
  pinMode(statusWifi, OUTPUT);
  ledWifi = true;
  digitalWrite(statusWifi, ledWifi);
  delay(5000);
  Serial.println();
  Serial.println();
  Serial.print("Conectando a red WiFi: ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    ledWifi = !ledWifi;
    digitalWrite(statusWifi, ledWifi);
    Serial.print(".");
  }
  Serial.println("");
  Serial.print("Conectado a red WiFi: ");
  Serial.println(ssid);
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  digitalWrite(statusWifi, LOW);
  /******SENSOR MAX30100******/
  ledWifi = true;
  digitalWrite(statusWifi, ledWifi);
  delay(2000);
  if (!pox.begin()) {
    Serial.println("OXIMETRO ERROR");
    ESP.restart();
    for(;;);
  } else {

```

```

    Serial.println("OXIMETRO OK");
}
pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
//pox.setOnBeatDetectedCallback(onBeatDetected);
/*****SENSOR MLX90614 *****/
if (!mlx.begin()) {
    Serial.println("Error de sensor MLX.");
    while (1);
};
Serial.print("Emissivity = "); Serial.println(mlx.readEmissivity());
Serial.println("=====");
);
    ledWifi = false;
    digitalWrite(statusWifi, ledWifi);
}

void loop(){
    count = 0;
    delay(2000);
    if (!pox.begin()) {
        Serial.println("FAILED");
        ESP.restart();
        for(;;);
    } else {
        Serial.println("SUCCESS");
    }
    Serial.println();
    pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
    Serial.println("Lectura de oximetro");
    tsLastReport = millis();
    while(true){
        pox.update();

```

```

if(millis() - tsLastReport > REPORTING_PERIOD_MS){
    valueFrecAux = pox.getHeartRate();
    valueOxiAux = pox.getSpO2();
    if(valueFrecAux>0.0 && valueOxiAux>0.0){
        Serial.println("--- " + String(count) + " ---");
        frecCardiaca[count] = valueFrecAux;
        Sop2[count] = valueOxiAux;
        Serial.println(frecCardiaca[count]);
        Serial.println(Sop2[count]);
        count++;
    }
    tsLastReport = millis();
}
if(count>10){
    Serial.println();
    frecMayor = frecCardiaca[1];
    for (int i = 1; i <= 10; i++) {
        Serial.print("|");
        Serial.print(frecCardiaca[i]);
        if (frecCardiaca[i] > frecMayor) {
            frecMayor = frecCardiaca[i]; // Actualizar si se encuentra un número mayor
        }
    }
    Serial.println();
    Serial.print("Frecuencia cardiaca: ");
    Serial.println(frecMayor);
    Serial.println("Oxigeno en la sangre:");
    for (int i = 0; i < 10; i++) {
        Serial.print("|");
        Serial.print(Sop2[i]);
        if(i >= 4){
            valueOxi = valueOxi + Sop2[i];
        }
    }
}

```

```

    }
    Serial.print("|: ");
    valueOxi = valueOxi/6;
    Serial.println(valueOxi);
    break;
}
}
Serial.print("lectura de temperatura");
count = 0;
tsLastReport = millis();
while(true){
    if(millis() - tsLastReport > REPORTING_PERIOD_MS){
        temp[count] = mlx.readObjectTempC();
        Serial.print(".");
        count++;
        tsLastReport = millis();
    }

    if(count>10){
        Serial.println();
        Serial.println("Temperatura corporal:");
        for (int i = 0; i < 10; i++) {
            Serial.print("|");
            Serial.print(temp[i]);
            if(i >= 4){
                valueTemp = valueTemp + temp[i];
            }
        }
        Serial.print("| : ");
        valueTemp = (valueTemp/6) + 2;
        Serial.println(valueTemp);
    }
}

```

```

Serial.println("=====
");
    break;
}
}
TagoHttpRequest(frecMayor, valueOxi, valueTemp);
ResetValue();
}

//Librería: Credential_secrets.h
/***** WIFI *****/
const char* ssid = "NOMBRE_DE_LA_RED_WIFI";
const char* password = "CONTRASEÑA";

//Función de reset
void ResetValue(){
    valueHR = 0.0;
    valueOxi = 0.0;
    valueTemp = 0.0;
    count = 0;
    for (int i = 0; i < 12; i++) {
        frecCardiaca[i] = 0.0;
        Sop2[i] = 0.0;
        temp[i] = 0.0;
    }
}

//Función IOT
// Define la URL y el TOKEN de tu cuenta TagoIO
const char* serverUrl = "https://api.tago.io/data";
const char* token = "1525d430-e6fb-48c9-9f5a-0c4e5f3fff12";
void TagoHttpRequest(float valueHR, float valueOxi, float valueTemp){
    String strHr = String(valueHR);

```

```

String strOxi = String(valueOxi);
String strTemp = String(valueTemp);
//String payload = "{\"Frecuencia\": " + strHr + ", \"Oxigeno\": " + strOxi + ",
\"Temperatura\": " + strTemp + "}";
String payload = "[\r\n";
    payload += "\"variable\": \"Frecuencia\", \r\n";
    payload += "\"unit\": \"LPM\", \r\n";
    payload += "\"value\":";
    payload += strHr;
    payload += "\r\n}, {\r\n";
    payload += "\"variable\": \"Oxigeno\", \r\n";
    payload += "\"unit\": \"%\", \r\n";
    payload += "\"value\":";
    payload += strOxi;
    payload += "\r\n}, {\r\n";
    payload += "\"variable\": \"Temperatura\", \r\n";
    payload += "\"unit\": \"C\", \r\n";
    payload += "\"value\":";
    payload += strTemp;
    payload += "\r\n}]\r\n";
Serial.println(payload);
// Crea una instancia del cliente HTTP
HTTPClient http;
// Configura la cabecera HTTP
http.begin(serverUrl);
http.addHeader("Content-Type", "application/json");
http.addHeader("Device-Token", token);
int httpResponseCode = http.POST(payload);
// Si la solicitud fue exitosa, imprime el código de respuesta
if (httpResponseCode > 0) {
    Serial.print("Código de respuesta HTTP: ");
    Serial.println(httpResponseCode);
} else {

```

```
Serial.print("Error en la solicitud HTTP: ");  
Serial.println(httpResponseCode);  
}  
  
// Libera los recursos  
http.end();  
}
```