



# **ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

## **FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

### **CARRERA TELECOMUNICACIONES**

# **"DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE ELECTROCARDIOGRAMO MÓVIL UTILIZANDO REDES MÓVILES PARA MONITORIZACIÓN DE LA FRECUENCIA CARDIACA EN TIEMPO REAL"**

## **Trabajo de Titulación**

Tipo: Dispositivo Tecnológico

Presentado para optar el grado académico de:

## **INGENIERO EN TELECOMUNICACIONES**

**AUTOR: GUILLERMO ALEJANDRO PINGUIL DUCHI**

Riobamba - Ecuador

2024



# **ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**CARRERA TELECOMUNICACIONES**

## **"DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE ELECTROCARDIÓGRAFO MÓVIL UTILIZANDO REDES MÓVILES PARA MONITORIZACIÓN DE LA FRECUENCIA CARDIACA EN TIEMPO REAL"**

**Trabajo de Titulación**

Tipo: Dispositivo Tecnológico

Presentado para optar el grado académico de:

**INGENIERO EN TELECOMUNICACIONES**

**AUTOR: GUILLERMO ALEJANDRO PINGUIL DUCHI**

**DIRECTOR: ING JEFFERSON RIBADENEIRA RAMÍREZ.**

Riobamba - Ecuador

2024

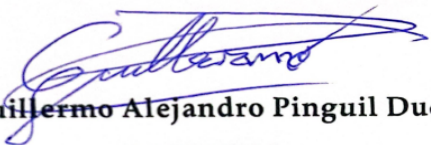
© 2024, **Guillermo Alejandro Pinguil Duchi**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, Guillermo Alejandro Pinguil Duchi, declaro que el presente Trabajo de Titulación es de mi autoría y los resultados de este son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor asumo la responsabilidad legal y académica de los contenidos de este trabajo de Titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 15 de abril del 2024.

  
**Guillermo Alejandro Pinguil Duchi**  
0302743125



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

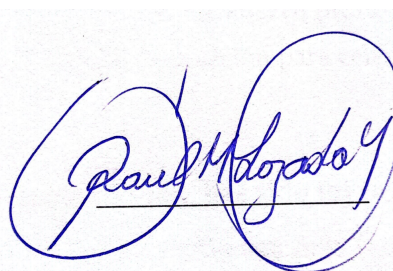
**CARRERA TELECOMUNICACIONES**

El Tribunal del Trabajo de Titulación certifica que: El Trabajo de Titulación Tipo: Dispositivo Tecnológico “DISEÑO E IMPLEMENTACION DE UN PROTOTIPO DE ELECTROCARDIOGRAFO MOVIL UTILIZANDO REDES MOVILES PARA LA MONITORIZACION DE LA FRECUENCIA CARDIACA EN TIEMPO REAL”, realizado por el señor: **GUILLERMO ALEJANDRO PINGUIL DUCHI**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

**FIRMA**

**FECHA**

Ing. Raúl Marcelo Lozada Yánez, MSc  
**PRESIDENTE DEL TRIBUNAL**



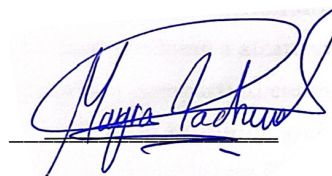
2024-04-15

Ing. Jefferson Ribadeneira Ramirez, PhD  
**DIRECTOR DE TRABAJO  
DE TITULACIÓN**



2024-04-15

Ing. Mayra Alejandra Pacheco Cunduri, MSc  
**ASESORA DEL TRABAJO  
DE TITULACIÓN**



2024-04-15

## DEDICATORIA

A mi madre, por su amor incondicional, su constante apoyo y su sacrificio para darme la oportunidad de perseguir mis sueños académicos. Tu amor incondicional han sido la fuerza motriz detrás de cada logro en mi vida. A través de tu constante aliento y orientación, me has enseñado el valor del esfuerzo y la perseverancia. Este logro no habría sido posible sin tu infinita paciencia y comprensión. Gracias por ser mi inspiración y por tu dedicación inquebrantable. Este trabajo está dedicado a ti, en reconocimiento a tu infinito amor y apoyo. Te amo más de lo que las palabras pueden expresar.

A mi querido hermano, quiero expresar mi profundo agradecimiento por todo lo que has representado para mí a lo largo de los años. Desde nuestros primeros días de infancia hasta el presente, has sido mucho más que un simple hermano; has sido mi amigo más cercano, mi confidente y mi compañero de vida. Tu presencia en mi existencia ha sido un regalo invaluable, y cada día contigo ha sido una verdadera bendición. Juntos, hemos atravesado los altibajos de la vida, compartiendo risas y lágrimas, y hemos construido un vínculo indestructible que supera cualquier adversidad. Agradezco profundamente tu apoyo, siendo mi roca en los momentos difíciles, y tu disposición para celebrar conmigo en los momentos de alegría.

Para ti que desde el momento en que entraste en mi vida, has sido mi mayor fuente de motivación y apoyo. Tu presencia ha hecho que cada desafío sea más fácil de superar y cada alegría sea aún más vibrante. Gracias por estar presente celebrando mis victorias y consolándome en los momentos difíciles. Tu comprensión, paciencia y cariño han sido un regalo invaluable que no puedo agradecer lo suficiente. Que esta dedicación sea un pequeño reflejo de mi inmenso amor y gratitud hacia ti.

A todas las personas y resto de familia gracias por estar siempre a mi lado, brindándome amor, comprensión y motivación. Cada uno de ustedes ha contribuido de manera invaluable a mi crecimiento personal y académico, inspirándome a alcanzar mis metas y superar desafíos. Este logro es también suyo, pues han compartido conmigo el viaje lleno de sacrificios y triunfos. Que esta dedicación refleje mi profunda gratitud y amor hacia cada uno de ustedes. Su apoyo ha sido un regalo inestimable que atesoraré siempre. ¡Gracias por ser mi familia y por ser mi mayor fuente de fortaleza y alegría!, de una forma u otra, han contribuido a este logro, mi más sincero agradecimiento.

Con todo mi cariño y gratitud.

## **AGRADECIMIENTO**

Quiero expresar mi sincero agradecimiento a mi querido hermano Pedro, cuya influencia y apoyo han sido una fuente constante de inspiración durante el proceso de investigación para esta tesis. Desde el inicio de este proyecto, ha sido una presencia motivadora y alentadora, brindándome su perspectiva única y compartiendo su conocimiento.

Quiero expresar mi sincero agradecimiento a mi tutor de tesis, por su guía experta, su dedicación incansable y su apoyo incondicional durante todo el proceso de investigación.

# ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS . . . . .	xii
ÍNDICE DE ILUSTRACIONES . . . . .	xv
ÍNDICE DE ECUACIONES . . . . .	xvi
ÍNDICE DE ANEXOS . . . . .	xvii
RESUMEN . . . . .	.xviii
SUMARY . . . . .	xix
INTRODUCCIÓN . . . . .	1
CAPÍTULO I	
1 <b>DIAGNÓSTICO DEL PROBLEMA</b>	2
1.1 <b>Planteamiento del problema</b> . . . . .	2
1.1.1 <i>Antecedentes</i> . . . . .	2
1.1.2 <i>Formulación del problema</i> . . . . .	3
1.1.2.1 <i>Sistematización del problema</i> . . . . .	3
1.2 <b>Justificación</b> . . . . .	4
1.2.1 <i>Justificación teórica</i> . . . . .	4
1.2.2 <i>Justificación aplicativa</i> . . . . .	5
1.3 <b>Alcance</b> . . . . .	7
1.4 <b>Objetivos</b> . . . . .	7
1.4.1 <i>Objetivo General</i> . . . . .	7
1.4.2 <i>Objetivos Específicos</i> . . . . .	7
CAPÍTULO II	
2 <b>MARCO TEÓRICO</b>	8
2.1 <b>EL corazón</b> . . . . .	8

2.1.1	<i>Las válvulas cardíacas</i>	9
2.1.2	<i>El sistema de conducción</i>	9
2.1.3	<i>El aparato circulatorio</i>	9
2.1.4	<i>Frecuencia cardíaca</i>	10
2.1.4.1	<i>Medición de frecuencia cardíaca</i>	10
2.1.5	<i>Electrocardiógrafo</i>	11
2.1.5.1	<i>TIPOS DE ELECTROCARDIOGRAFOS</i>	11
2.2	<b>Sensores para detectar frecuencia cardíaca</b>	13
2.2.1	<i>Sensor Pulso</i>	13
2.2.2	<i>Sensor de frecuencia cardíaca ECG AD8232</i>	15
2.3	<b>Redes inalámbricas</b>	17
2.3.1	<i>Arquitectura de redes inalámbricas</i>	18
2.3.1.1	<i>Modo Ad hoc</i>	18
2.3.1.2	<i>Modo Infraestructura</i>	18
2.3.2	<b>Bluetooth</b>	19
2.3.2.1	<i>Clases de Bluetooth</i>	19
2.3.2.2	<i>Versiones de Bluetooth</i>	20
2.3.2.3	<i>Módulos bluetooth</i>	21
2.3.2.4	<i>Módulo HC-06</i>	22
2.3.2.5	<i>Funciones</i>	22
2.4	<b>Comunicaciones móviles</b>	23
2.4.0.1	<i>Cuarta generación 4G</i>	24
2.4.0.2	<i>Quinta generación 5G</i>	25
2.5	<b>Tarjetas de desarrollo</b>	26
2.5.1	<i>Arduino</i>	26
2.5.2	<i>RASPBERRY PI</i>	27
2.5.3	<i>ESP32</i>	28

2.5.4	<i>PHOTON</i>	29
2.6	<b>Creación de aplicaciones móviles</b>	30
2.6.1	<i>NATIVAS</i>	30
2.6.2	<i>HIBRIDAS</i>	30
2.6.3	<i>MULTIPLATAFORMA</i>	30
2.6.4	<i>WEB/HTML5</i>	31
2.6.5	<i>APP INVENTOR</i>	31
2.6.6	<i>ANDROID STUDIO</i>	31
2.7	<b>SERVICIOS CLOUDING</b>	33
2.7.1	<b><i>FUNCIONAMIENTO DE LOS SERVICIOS DE NUBE (FIREBASE)</i></b>	33
2.7.1.1	<i>BaaS</i>	34
2.7.1.2	<i>Realtime Database</i>	34
2.7.2	<b><i>INFRAESTRUCTURA DE LA NUBE</i></b>	34

### CAPÍTULO III

3	<b>MARCO METODOLÓGICO</b>	<b>36</b>
3.1	<b>Requerimientos del sistema</b>	<b>36</b>
3.1.1	<i>Requerimientos del hardware del sistema</i>	36
3.1.2	<i>Requerimientos del software del sistema</i>	36
3.2	<b>Descripción general del sistema</b>	<b>37</b>
3.3	<b>Concepción del sistema</b>	<b>38</b>
3.3.1	<i>Metodología para el desarrollo del prototipo (Mockup)</i>	38
3.4	<b>Arquitectura del sistema</b>	<b>39</b>
3.4.1	<i>Nodo transmisor</i>	39
3.4.1.1	<i>Tarjeta de desarrollo (Arduino UNO)</i>	42
3.4.1.2	<i>Sensor cardíaco AD8232</i>	43
3.4.1.3	<i>Módulo Bluetooth HC-06</i>	46

3.4.1.4	<i>Fuente de alimentación</i>	48
3.4.2	<b>Nodo receptor</b>	52
3.4.2.1	<i>Aplicación móvil</i>	53
3.4.2.2	<i>Página web</i>	55
3.5	<b>Toma de resultados</b>	56

## CAPÍTULO IV

4	<b>RESULTADOS</b>	<b>59</b>
4.1	<b>Etapa 1: Diseño</b>	59
4.1.1	<i>Diseño de la placa y conexiones</i>	59
4.1.2	<i>Diseño de la aplicación móvil</i>	60
4.1.3	<i>Diseño de la página web</i>	64
4.2	<b>Etapa 2: Diseño del protector plástico</b>	68
4.3	<b>Etapa 3: Montaje del prototipo</b>	69
4.4	<b>Etapa 4: Validación del prototipo</b>	72
4.5	<b>VALIDACIÓN DE PROTOTIPO</b>	73
4.6	<b>Recopilación de mediciones con el prototipo</b>	73
4.7	<b>Recopilación de mediciones con un electrocardiógrafo de mercado</b>	78
4.8	<b>Comparativa del prototipo con el electrocardiógrafo</b>	79
4.8.1	<i>Comparativa del prototipo con el electrocardiógrafo</i>	79
4.8.2	<i>Latencia de la transmisión de la información desde el prototipo (módulo Bluetooth) a la aplicación móvil</i>	84
4.8.3	<i>Latencia de la transmisión de la información desde la aplicación móvil a la página web considerando el modo de conexión, ya sea por LTE o WiFi</i>	85
4.9	<b>Análisis económico</b>	87

## CAPÍTULO V

5	<b>CONCLUSIONES Y RECOMENDACIONES</b>	<b>89</b>
---	---------------------------------------	-----------

5.1	Conclusiones . . . . .	89
5.2	Recomendaciones . . . . .	90

**BIBLIOGRAFÍA**

**ANEXOS . . . . .**



## ÍNDICE DE TABLAS

<b>Tabla 2-1:</b> Tabla comparativa entre sensores . . . . .	16
<b>Tabla 3-1:</b> Tabla comparativa entre sensores . . . . .	43
<b>Tabla 3-2:</b> Descripción de Pines del sensor AD8232 . . . . .	44
<b>Tabla 3-3:</b> Descripción de Pines y componentes del módulo Bluetooth HC-06 . . .	47
<b>Tabla 3-4:</b> Descripción de Pines y componentes del módulo Bluetooth HC-06 . . .	50
<b>Tabla 3-5:</b> Comandos de la aplicación móvil . . . . .	55
<b>Tabla 4-1:</b> Análisis económico . . . . .	88

## ÍNDICE DE ILUSTRACIONES

<b>Ilustración 1-1:</b>	Sistema de comunicación electrocardiógrafo Bluetooth, de autoría propia. . . . .	5
<b>Ilustración 1-2:</b>	Prototipo electrocardiógrafo Bluetooth, de autoría propia. . . .	5
<b>Ilustración 1-3:</b>	Receptor bluetooth Android, de autoría propia. . . . .	6
<b>Ilustración 1-4:</b>	Receptor bluetooth Android, de autoría propia. . . . .	6
<b>Ilustración 2-1:</b>	Corazón . . . . .	8
<b>Ilustración 2-2:</b>	Electrocardiógrafo mono canal . . . . .	12
<b>Ilustración 2-3:</b>	Electrocardiógrafo multi canal . . . . .	12
<b>Ilustración 2-4:</b>	Electrocardiógrafo multi canal con impresión . . . . .	13
<b>Ilustración 2-5:</b>	Sensor de pulso . . . . .	14
<b>Ilustración 2-6:</b>	ECG AD8232 . . . . .	15
<b>Ilustración 2-7:</b>	Clasificación de las redes inalámbricas . . . . .	17
<b>Ilustración 2-8:</b>	Modulo HC-06 . . . . .	22
<b>Ilustración 2-9:</b>	Arduino UNO . . . . .	27
<b>Ilustración 2-10:</b>	Raspberry PI . . . . .	28
<b>Ilustración 2-11:</b>	ESP32 . . . . .	29
<b>Ilustración 2-12:</b>	PHOTON . . . . .	29
<b>Ilustración 2-13:</b>	APP inventor . . . . .	31
<b>Ilustración 2-14:</b>	Android studio . . . . .	32
<b>Ilustración 2-15:</b>	Servicios Clouding . . . . .	33
<b>Ilustración 2-16:</b>	Infraestructura de nube . . . . .	35
<b>Ilustración 3-1:</b>	Diagrama de bloques del sistema. . . . .	38
<b>Ilustración 3-2:</b>	Metodología Mockup. . . . .	39
<b>Ilustración 3-3:</b>	Diagrama de bloques del sistema, nodo transmisor lector . . .	40
<b>Ilustración 3-4:</b>	Diagrama de flujo, nodo transmisor lector . . . . .	41
<b>Ilustración 3-5:</b>	Arduino UNO . . . . .	42
<b>Ilustración 3-6:</b>	Conexiones del sensor AD8232 . . . . .	45
<b>Ilustración 3-7:</b>	Declaración de variables . . . . .	45
<b>Ilustración 3-8:</b>	Bloque void setup . . . . .	46
<b>Ilustración 3-9:</b>	Bloque void loop . . . . .	46
<b>Ilustración 3-10:</b>	Conectividad del módulo Bluetooth . . . . .	47
<b>Ilustración 3-11:</b>	Código del módulo Bluetooth . . . . .	48
<b>Ilustración 3-12:</b>	Batería lipo. . . . .	49
<b>Ilustración 3-13:</b>	Pines del módulo TP4056 . . . . .	50

<b>Ilustración 3-14:</b> Convertidor MT3608 . . . . .	50
<b>Ilustración 3-15:</b> Conexiones del prototipo con la Batería lipo. . . . .	51
<b>Ilustración 3-16:</b> Esquema electrónico . . . . .	52
<b>Ilustración 3-17:</b> Diagrama de bloques del sistema, nodo transmisor receptor . .	52
<b>Ilustración 3-18:</b> Diagrama de bloques del sistema, módulo de registro . . . . .	53
<b>Ilustración 3-19:</b> Diagrama de flujo del sistema, módulo de registro. . . . .	54
<b>Ilustración 3-20:</b> Diagrama de flujo de la página web. . . . .	56
<b>Ilustración 3-21:</b> Esquema general del prototipo. . . . .	57
<b>Ilustración 3-22:</b> Diagrama de bloque de la toma de resultados. . . . .	58
<b>Ilustración 4-1:</b> Diseño PCB . . . . .	60
<b>Ilustración 4-2:</b> Visualización 3D . . . . .	60
<b>Ilustración 4-3:</b> Componentes de la aplicación móvil . . . . .	61
<b>Ilustración 4-4:</b> Activación de Bluetooth . . . . .	61
<b>Ilustración 4-5:</b> Conectividad Módulo Bluetooth . . . . .	62
<b>Ilustración 4-6:</b> Mensaje de conectividad . . . . .	62
<b>Ilustración 4-7:</b> Datos en tiempo real . . . . .	63
<b>Ilustración 4-8:</b> Datos en tiempo real . . . . .	63
<b>Ilustración 4-9:</b> Mensajes de verificación de envío a la nube . . . . .	64
<b>Ilustración 4-10:</b> Área de visualización . . . . .	64
<b>Ilustración 4-11:</b> Seguridad página web . . . . .	65
<b>Ilustración 4-12:</b> Plataforma Firebase . . . . .	66
<b>Ilustración 4-13:</b> Credenciales cargadas en Firebase . . . . .	66
<b>Ilustración 4-14:</b> Proceso de nuevas credenciales en Firebase . . . . .	67
<b>Ilustración 4-15:</b> Componentes página web . . . . .	68
<b>Ilustración 4-16:</b> Mediciones de la carcasa . . . . .	69
<b>Ilustración 4-17:</b> Mediciones de la tapa de la carcasa . . . . .	69
<b>Ilustración 4-18:</b> Placa impresa vista superior . . . . .	70
<b>Ilustración 4-19:</b> Placa impresa vista inferior . . . . .	70
<b>Ilustración 4-20:</b> Montaje del prototipo (vista frontal) . . . . .	71
<b>Ilustración 4-21:</b> Montaje del prototipo (vista posterior) . . . . .	71
<b>Ilustración 4-22:</b> Ubicación de los electrodos . . . . .	72
<b>Ilustración 4-23:</b> Paciente 1 . . . . .	73
<b>Ilustración 4-24:</b> : Resultados del Paciente 1 (página web) . . . . .	74
<b>Ilustración 4-25:</b> : Paciente 2 . . . . .	74
<b>Ilustración 4-26:</b> : Resultados del Paciente 2 (página web) . . . . .	75
<b>Ilustración 4-27:</b> Paciente 3 . . . . .	75
<b>Ilustración 4-28:</b> Resultados del Paciente 3 (página web) . . . . .	76

<b>Ilustración 4-29:</b> Paciente 4 . . . . .	76
<b>Ilustración 4-30:</b> Resultados del Paciente 4 (página web) . . . . .	77
<b>Ilustración 4-31:</b> Paciente 5 . . . . .	77
<b>Ilustración 4-32:</b> Resultados del Paciente 5 (página web) . . . . .	78
<b>Ilustración 4-33:</b> Mediciones del Paciente 1 . . . . .	78
<b>Ilustración 4-34:</b> : Mediciones del Paciente 2 . . . . .	79
<b>Ilustración 4-35:</b> Comparación de mediciones paciente 1 . . . . .	80
<b>Ilustración 4-36:</b> Comparación de mediciones paciente 2 . . . . .	81
<b>Ilustración 4-37:</b> Comparación de mediciones paciente 3 . . . . .	82
<b>Ilustración 4-38:</b> Comparación de mediciones paciente 4 . . . . .	83
<b>Ilustración 4-39:</b> Comparación de mediciones paciente 5 . . . . .	84
<b>Ilustración 4-40:</b> Resultados de latencia (Bluetooth) . . . . .	85
<b>Ilustración 4-41:</b> Resultados de latencia (LTE) . . . . .	86
<b>Ilustración 4-42:</b> Resultados de latencia (Wi-Fi) . . . . .	87

## ÍNDICE DE ECUACIONES

## **ÍNDICE DE ANEXOS**

**ANEXO A:** CODIGO ELECTROCARDIOGRAFO BLUETOOTH (IDE ARDUINO)

**ANEXO B:** CODIGO APLICACION MÓVIL

**ANEXO C:** CODIGO PAGINA WEB

**ANEXO D:** ENTREVISTA AL MÉDICO

## RESUMEN

Los problemas cardiacos son una de las principales causas de mortalidad en todo el mundo, contribuyendo a un gran número de hospitalizaciones y muertes prematuras. La detección temprana y el monitoreo facilitado de este órgano son necesarios para prevenir complicaciones graves y mejorar el pronóstico de los pacientes con problemas cardíacos. Por lo tanto, el objetivo de la presente investigación es desarrollar un prototipo de electrocardiógrafo móvil que tenga la capacidad de monitorear al corazón de forma fácil y permita guardar esta información en una base de datos a la cual se pueda acceder usando internet y redes móviles. En la metodología, se utilizaron métodos artesanales para la fabricación de un dispositivo con la capacidad de captar las señales eléctricas del corazón para luego ser enviadas a un smartphone Android utilizando la tecnología Bluetooth. Esta información es administrada por una aplicación desarrollada con el lenguaje de programación Kotlin para luego ser enviada a una base de datos en Firebase para que finalmente esta información sea interpretada usando una página web desarrollada en lenguaje HTML. Mediante el uso de esta metodología se logró producir un prototipo que tiene la capacidad de obtener la gráfica del comportamiento eléctrico del órgano cardiaco. En este contexto, se logró crear un prototipo y sistema que permite monitorear en tiempo real, determinar comportamientos anormales del corazón, subir la información a una base de datos y acceso a esta información utilizando internet.

**Palabras clave:** <MORTALIDAD>, <MONITOREO>, <PROTOTIPO>, <ELECTROCARDIÓGRAFO>, <MÓVIL>, <CORAZÓN>, <COMPORTAMIENTO>.  
0444-DBRA-UPT-2024

## SUMMARY

Cardiac issues are among the leading causes of mortality worldwide, contributing to a significant number of hospitalizations and premature deaths. Early detection and heart monitoring are essential to prevent serious complications and improve the prognosis for patients with cardiac problems. Therefore, this research aims to develop a prototype of a mobile electrocardiograph to monitor the heart effortlessly, allowing the information in a database accessible via the internet and mobile networks. The methodology used traditional methods to manufacture a device capable of capturing the heart's electrical signals, which are then transmitted to an Android smartphone using Bluetooth technology. This data is managed by an application developed in the Kotlin programming language and subsequently sent to a Firebase database. Finally, this information is interpreted using a webpage developed in HTML, through this methodology, a prototype was successfully produced that is capable of obtaining the graphical representation of the heart's electrical activity. In this context, a prototype and a system were created that allow real-time monitoring, determination of abnormal heart behaviors, uploading information to a database, and access to this information using the internet.

**KEYWORDS:** <MORTALITY>, <MONITORING>, <PROTOTYPE>, <ELECTROCARDIOGRAPH>, <MOBILE>, <HEART>, <BEHAVIOR>.



Lic. Maritza Larrea Mg.  
0603370784



## INTRODUCCIÓN

Actualmente la tecnología se desarrolla a pasos agigantados permitiendo facilitar el día a día de las personas, la tecnología es utilizada en un sin fin de ramas, una de estas es la medicina ayudando a mejorar ciertos procesos y facilitando la vida de muchos pacientes que sufren de distintos tipos de enfermedades, entre estas los problemas cardiacos, normalmente estos pacientes tienen que recurrir con frecuencia a controles médicos afectando en gran medida su calidad de vida, tomando en cuenta este problema es necesario poder controlar la frecuencia cardiaca de una persona para poder controlar el estado de su corazón, pero este tipo de controles tienen que ser realizados en hospitales o en lugares muy específicos, limitando la capacidad de los médicos para poder diagnosticar con certeza el tipo de afección, y que esta pueda ser tratada de forma correcta y a tiempo.

En pacientes con problemas cardiacos suele ser muy común la dificultad para detectar anomalías, debido a que estas solo se presentan en momentos específicos, y puede pasar que en el momento que se realizan los estudios no se detecta ninguna anomalía, haciendo que este tipo de pacientes no se sometan a los tratamientos pertinentes, provocando que sus corazones se sigan deteriorando. Esto hace que sea necesario poder realizar el monitoreo de la frecuencia cardiaca de una persona en el momento exacto que está sufriendo algún tipo de alteración en el comportamiento normal de su corazón, es necesario que este monitoreo se pueda hacer en tiempo real y de forma remota; también toda la información debe estar al alcance de un médico a través de una base de datos en la cual se guarde toda esta información. Para facilitar el acceso a la información es necesario poder acceder a estos registros de forma remota para facilitar el análisis de la frecuencia cardiaca de una persona, ayudando en el diagnóstico de la afección que pueda llegar a tener un paciente.

Dando tratamiento a la problemática y necesidad de monitorear la frecuencia cardiaca de una persona de forma remota, se plantea el diseño e implementación de un prototipo de electrocardiógrafo móvil utilizando redes móviles para la monitorización de la frecuencia cardiaca en tiempo real, el cual consta de un conjunto de componentes electrónicos como tarjetas de desarrollo, módulos bluetooth sensores capaces de detectar la frecuencia cardiaca, todo esto integrado con dispositivo móvil que permita observar el correcto funcionamiento, conectividad a una red móvil y la capacidad de enviar todos los datos a una nube para facilitar el acceso a la información desde cualquier lugar siempre y cuando se cuente con acceso a internet.

## CAPÍTULO I

### 1. DIAGNÓSTICO DEL PROBLEMA

El capítulo I se basa en la recolección de información referencial para el correcto desarrollo del prototipo, en este se detallarán aspectos de diseño, metodología, así como antecedentes de proyectos que puedan llegar a tener información sustancial para el proyecto.

#### 1.1 Planteamiento del problema

##### 1.1.1 Antecedentes

En la actualidad gracias al avance tecnológico de las comunicaciones móviles, varios sectores han revolucionado la forma de abordar ciertas actividades, incluido la rama de la salud. La monitorización de forma remota de signos vitales se ha convertido en una herramienta indispensable para la detección eficaz y a tiempo de problemas de salud y el seguimiento de pacientes en tiempo real.

En el departamento de Telecomunicaciones, Facultad de Ingeniería en Telecomunicaciones, Informática y Biomédica, Universidad de Oriente, Santiago de Cuba, Cuba, en el año 2020 se propone la tesis: “Diseño de un sistema inalámbrico de monitorización electrocardiográfica para dispositivos Android, utilizando la tecnología wifi” (SMITH et al., 2020). “En este trabajo se describe el diseño e implementación de un sistema inalámbrico de monitorización electrocardiográfica para dispositivos Android. El sistema está formado por un dispositivo que actuará como servidor ubicado en el paciente y otro dispositivo que actuará como cliente. El servidor incorpora las etapas de protección contra desfibriladores externos, adquisición y transmisión de la señal mediante un módulo Wi-Fi (ESP8266 NodeMCU), y el cliente recibe, procesa y visualiza en tiempo real la señal de electrocardiograma (ECG) en un teléfono inteligente o en una tableta electrónica, con el sistema operativo Android” (SMITH et al., 2020).

En el departamento de electrónica y comunicaciones, Amaljyothi, escuela de ingeniería, propone el “Diseño de un sistema de monitoreo de electrocardiograma móvil basado en Wi-Fi en la plataforma Concerto” (AHAMMED et al., 2013). En este proyecto se desarrolla un “sistema de monitoreo de ECG inalámbrico que integra tecnología Wi-Fi. Este sistema basado en Wi-Fi se compone de un módulo de adquisición de señales de ECG de un solo chip en Concerto MCU, un módulo Wi-Fi SimpleLink CC3000 y un teléfono inteligente,

La velocidad de datos del sistema es de 11 Mbps como mínimo y de 54 Mbps como máximo. El módulo Wi-Fi utiliza una red de modo de infraestructura de IEEE802.11. El iPhone 4S de Apple se selecciona como la plataforma de dispositivo móvil, que incorpora Wi-Fi e iOS. En este documento, el sistema de monitoreo puede adquirir señales de ECG a través de un sensor de electrocardiograma de 2 derivaciones, transmitir los datos de ECG a través del enlace inalámbrico, procesar y mostrar la forma de onda de ECG en un teléfono inteligente” (AHAMMED et al., 2013).

En la Universidad Politécnica Salesiana de la ciudad de Cuenca se desarrolla la tesis denominada "Diseño e implementación de un electrocardiógrafo ambulatorio dinámico de una derivación con conexión inalámbrica a teléfono móvil inteligente y envío de información vía SMS o GPRS"(MATUTE et al., 2012). Este proyecto de investigación tiene como objeto "la captura de las señales del corazón por medio de 3 electrodos colocados en la zona precordial, estas señales son procesadas de manera analógica y digital para luego ser enviadas mediante Bluetooth al teléfono inteligente"(MATUTE et al., 2012) .

En la actualidad gracias al desarrollo de las tecnologías y redes es posible enviar datos o multimedia a grandes velocidades posibilitando video llamadas, llamadas o simplemente intercambio de información de forma prácticamente instantánea, dando la posibilidad de hacer un monitoreo de tiempo real a cualquier tipo de sensor que este subiendo su información a la nube, posibilitando estar pendiente a cambios que sufra los datos monitoreados sin necesidad de estar presente en la zona geográfica donde se realiza el monitoreo.

### ***1.1.2 Formulación del problema***

¿Como diseñar e implementar un prototipo de un dispositivo de bajo costo que permita monitorear la actividad cardiaca de forma remota y sencilla?

#### ***1.1.2.1 Sistematización del problema***

¿Como obtener información de sensores biométricos capaces de detectar los pulsos generados por el corazón?

¿Cómo planifica un sistema que permita detectar las señales eléctricas generadas por el corazón para que puedan ser enviadas de forma inalámbrica?

¿Cómo obtener las señales eléctricas generadas por el corazón para que sean enviadas de forma inalámbrica?

¿Cómo establecer una comunicación que permita llevar los datos recopilados desde el corazón hacia la nube y visualización de datos en nube?

¿Cómo evaluar el funcionamiento del prototipo y la red de comunicaciones?

## **1.2 Justificación**

### **1.2.1 Justificación teórica**

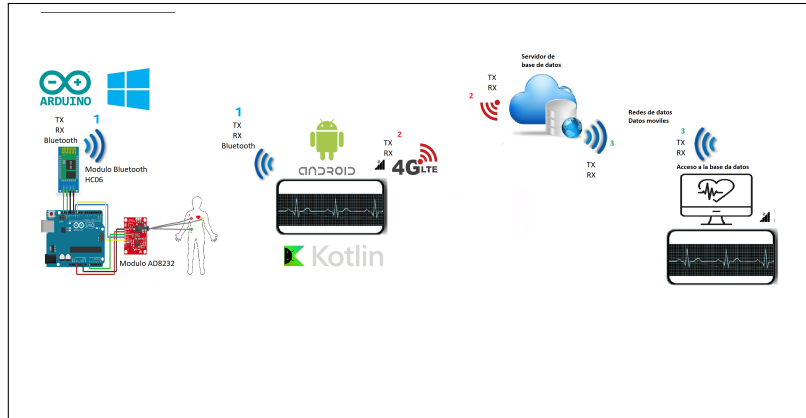
Según Zavala “El electrocardiograma no es más que un conjunto de ondas que Einthoven denominó P, Q, R, S, T y U de acuerdo con el orden de aparición en el tiempo. Recordando que la onda P se inscribe como resultado de la activación auricular y que, de inmediato, aparecen Q, R y S, integrando el complejo ventricular por la propagación de la onda de excitación a la musculatura de ambos ventrículos y al tabique interventricular”(ZAVALA, 2013). Este proporciona a los médicos información importante de la actividad cardiaca de una persona, permitiendo hacer valoraciones del correcto o incorrecto funcionamiento del corazón, sin embargo, este examen debe ser realizado en ambientes especializados y por personas con el conocimiento necesario para poder ser interpretado correctamente.

Las características y datos arrojados por un electrocardiograma son de suma importancia, estos datos permiten realizar diagnósticos a tiempo para poder salvaguardar la vida de los seres humanos, por lo tanto, es de gran interés el desarrollo de dispositivos capaces de monitorear las señales cardiacas. Además, se busca mejorar la calidad de vida de las personas que sufren de enfermedades cardiacas, tratando de evitar visitas tan frecuentes a casas de salud y sobre todo mejorar la capacidad de monitorear constantemente a este tipo de personas.

En la actualidad se utiliza constantemente tecnologías para transmitir información, muchas de estas tienen la virtud de ser inalámbricas y portátiles. Además, brindan grandes capacidades en cuanto a su velocidad de transmisión y ancho de banda posibilitando enviar cualquier tipo de información. Así mismo, esto hace posible el intercambio de información entre varios dispositivos dando lugar a la posibilidad de que un electrocardiograma tenga la capacidad de utilizar una red de comunicaciones inalámbricas que permita el acceso remoto y facilite el monitoreo en tiempo real de los pulsos emitidos por el corazón.

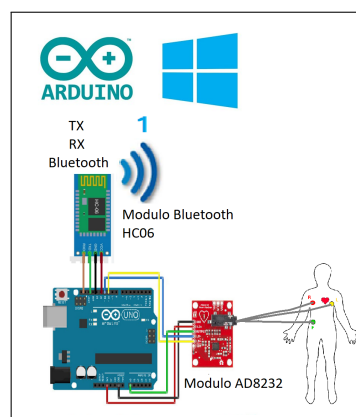
### 1.2.2 Justificación aplicativa

En la Ilustración 1-1 se muestra el sistema completo, donde se inicia la recopilación de datos usando el electrocardiógrafo para ser enviados usando la tecnología inalámbrica Bluetooth. Estos datos son receptados por el dispositivo móvil para que esta información sea redireccionada a la nube y pueda ser visualizada de forma sencilla y global.



**Ilustración 1-1:** Sistema de comunicación electrocardiógrafo Bluetooth, de autoría propia.

En la Ilustración 1-2 el receptor bluetooth grafica la señal recibida, para que el dispositivo móvil pueda mostrar la gráfica es necesario desarrollar una aplicación que sea capaz de recibir estos datos y mostrarlos en tiempo real, luego el dispositivo movil utilizando la tecnología de red móvil (LTE) envía la información a una base de datos en nube facilitando el acceso remoto a la información.



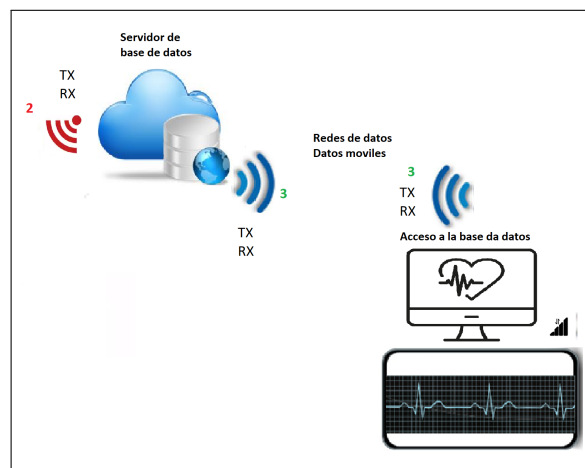
**Ilustración 1-2:** Prototipo electrocardiógrafo Bluetooth, de autoría propia.

En la Ilustración 1-3 el receptor bluetooth grafica la señal recibida, para que el dispositivo móvil pueda mostrar la gráfica es necesario desarrollar una aplicación, la cual sea capaz de recibir estos datos y mostrarlos en tiempo real, de esta forma se puede asegurar el correcto funcionamiento del electrocardiógrafo para que el dispositivo utilice su tecnología de red móvil (LTE) y envíe toda esta información a una base de datos en nube facilitando el acceso a esta información.



**Ilustración 1-3:** Receptor bluetooth Android, de autoría propia.

La información contenida en la nube puede ser visualizada desde cualquier lugar geográfico siempre que se cuente con acceso a la red, en la Ilustración 1-4 se describe la facilidad que existe para acceder a los datos guardados, pudiendo acceder a esta información desde cualquier dispositivo móvil u ordenador.



**Ilustración 1-4:** Receptor bluetooth Android, de autoría propia.

### **1.3 Alcance**

El trabajo tiene como finalidad el diseño y desarrollo de un prototipo de un dispositivo que tenga la capacidad de obtener el Electrocardiograma de una persona de forma remota y móvil, y que pueda ser usado en el instante que sea necesario la obtención de la frecuencia cardiaca de un paciente sin necesidad de acudir a un centro de salud.

### **1.4 Objetivos**

#### ***1.4.1 Objetivo General***

Diseñar e implementar un prototipo de electrocardiógrafo móvil utilizando redes móviles para monitorización de la frecuencia cardiaca en tiempo real.

#### ***1.4.2 Objetivos Específicos***

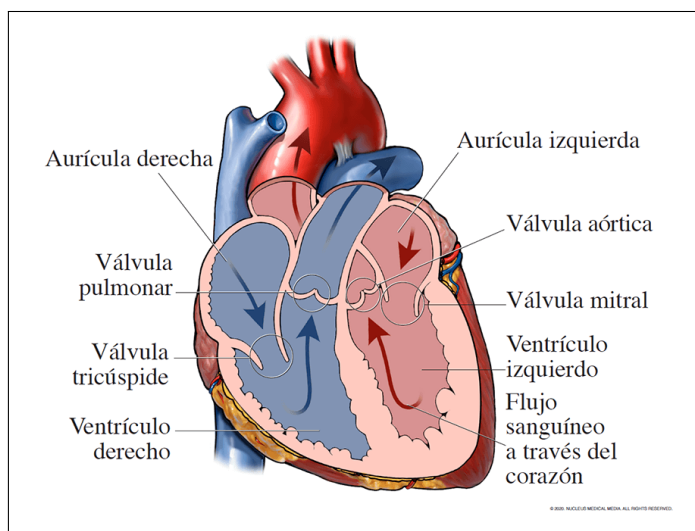
- Diseñar un prototipo que permita monitorear los pulsos eléctricos del corazón para enviar estos datos de forma inalámbrica y almacenarlos en Nube.
- Implementar un prototipo que permita monitorear los pulsos eléctricos del corazón, usando sensores biométricos y módulos bluetooth
- Diseñar interfaces de usuario para dispositivos móviles (Android) que permita manejar los datos en la nube.
- Evaluar el funcionamiento del prototipo con mediciones de tráfico y comparativa del dispositivo con un electrocardiógrafo comercial.

## CAPÍTULO II

### 2. MARCO TEÓRICO

#### 2.1 EL corazón

El corazón ocupa una posición central en el pecho, entre los pulmones, ubicándose detrás y ligeramente hacia la izquierda del esternón. Está envuelto por una cubierta de dos capas conocida como “pericardio”, actuando como una envoltura. La capa exterior del pericardio rodea la base de los principales conductos sanguíneos que alimentan al corazón, conectándose a la columna vertebral, al diafragma y otras áreas del cuerpo a través de ligamentos. La capa interna del pericardio se adhiere al músculo cardíaco. Una película de líquido separa ambas capas de esta membrana, permitiendo que el corazón se mueva al latir mientras permanece conectado al cuerpo.



**Ilustración 2-1: Corazón**

Fuente: (MEDIOS-MEDICOS-DEL-NUCLEO, 2008)

Según la Ilustración 2-1 el corazón cuenta con cuatro compartimentos. Los dos superiores se conocen como “aurícula izquierda” y “aurícula derecha”, mientras que los dos inferiores se llaman “ventrículo izquierdo” y “ventrículo derecho”. Una barrera muscular llamada “tabique” separa las aurículas izquierda y derecha, así como los ventrículos izquierdo y derecho. Destacando entre ellos, el ventrículo izquierdo que se presenta como la cavidad más grande y poderosa del corazón. A pesar de tener una pared con un grosor de aproximadamente media pulgada (un poco más de un centímetro), tiene la fuerza necesaria para impulsar la sangre a través de la válvula aórtica hacia todo el cuerpo.



### **2.1.1 Las válvulas cardíacas**

El control del flujo sanguíneo a través del corazón se realiza mediante cuatro válvulas distintas:

- La válvula tricúspide gestiona el flujo de sangre entre la aurícula y el ventrículo derechos.
- La válvula pulmonar regula el paso de la sangre desde el ventrículo derecho hacia las arterias pulmonares, encargadas de dirigir la sangre a los pulmones para su oxigenación.
- La válvula mitral facilita el paso de la sangre rica en oxígeno desde la aurícula izquierda hacia el ventrículo izquierdo, proveniente de los pulmones.
- La válvula aórtica permite que la sangre oxigenada pase desde el ventrículo izquierdo hacia la aorta, la arteria más grande del cuerpo, encargada de distribuir la sangre oxigenada a todo el organismo.

### **2.1.2 El sistema de conducción**

Los latidos del corazón son el resultado de impulsos eléctricos generados por el músculo cardíaco, el miocardio. Estos impulsos se originan en el nódulo sinoauricular (SA), situado en la parte superior de la aurícula derecha y conocido como el "marcapasos natural" del corazón. La señal eléctrica del SA se difunde por las fibras musculares de las aurículas y los ventrículos, induciendo su contracción. Aunque el nódulo SA emite impulsos a una velocidad constante, la frecuencia cardíaca puede variar debido a demandas físicas, niveles de estrés o factores hormonales.

### **2.1.3 El aparato circulatorio**

El sistema cardiovascular, integrado por el corazón y el sistema circulatorio, funciona como un sistema vital,<sup>E1</sup> corazón desempeña el papel de una bomba impulsora de la sangre hacia órganos, tejidos y células en todo el cuerpo. La sangre suministra oxígeno y nutrientes a cada célula y recoge el dióxido de carbono y las sustancias de desecho producidas por esas células. La sangre es transportada desde el corazón al resto del cuerpo por medio de una red compleja de arterias, arteriolas y capilares y regresa al corazón por las vénulas y venas. Si se unieran todos los vasos de esta extensa red y se colocaran en línea

recta, cubrirían una distancia de 60.000 millas (más de 96.500 kilómetros), lo suficiente como para circundar la tierra más de dos veces"(ANATOMIA DEL CORAZON, 2023).

#### **2.1.4 Frecuencia cardiaca**

La frecuencia cardiaca representa el número de veces que el corazón se contrae en un lapso de un minuto. Mantener este ritmo dentro del rango considerado adecuado por los profesionales de la salud es esencial para asegurar el funcionamiento óptimo del corazón. Cualquier desviación de esta frecuencia puede indicar un problema potencial. Esta tasa de latidos por minuto está influenciada por una variedad de factores, como la genética, la edad, el nivel de actividad física, el estrés y otros más. Por esta razón, la frecuencia cardiaca puede variar significativamente de una persona a otra, e incluso dentro del mismo individuo en diferentes momentos del día. Se ve afectada, por ejemplo, por las emociones, que pueden aumentarla, o durante el sueño, cuando disminuyen las necesidades de oxígeno del cuerpo.

Existen circunstancias que pueden elevar la frecuencia cardiaca, aunque algunas no representan un motivo de preocupación. Actividades como el ejercicio físico o estados de nerviosismo pueden llevar los latidos por encima de 100 por minuto. En ciertas condiciones médicas, como anemia o trastornos tiroideos, superar este umbral también puede ser considerado normal. Sin embargo, si la frecuencia cardiaca excede los 100 latidos por minuto sin que se presenten estas circunstancias, es recomendable buscar asesoramiento médico. Un especialista puede realizar un electrocardiograma para determinar la frecuencia cardiaca exacta y evaluar si existen irregularidades en el ritmo cardíaco, así como investigar otras posibles causas, como anemia, trastornos tiroideos o el consumo de estimulantes como cafeína o sustancias tóxicas.

##### **2.1.4.1 Medición de frecuencia cardiaca**

La medición de la frecuencia cardiaca se realiza mediante la palpación del pulso en la muñeca. Se indica colocar suavemente los dedos índice y medio aproximadamente entre uno y dos centímetros debajo del pliegue de la muñeca, entre el hueso perceptible en el exterior y el tendón que se encuentra en la parte interna. En esa posición, se ejerce una leve presión en la parte interna de la muñeca, cerca del dedo pulgar.

Debido a que la frecuencia cardiaca varía según la actividad, se recomienda realizar esta medición estando sentado, en reposo y en un entorno con temperatura moderada. También es aconsejable abstenerse de consumir cafeína u otros estimulantes antes de realizar la

medición.

Para mantener una frecuencia cardiaca basal óptima, los expertos sugieren la práctica regular de ejercicio físico. El ejercicio continuado es fundamental para mejorar la salud cardiovascular y física, controlar el peso corporal y reducir los riesgos asociados al corazón. A través del ejercicio, se logra regular la actividad del sistema nervioso simpático, lo que conduce a una disminución progresiva en la frecuencia cardíaca, proporcionando beneficios a largo plazo para el corazón.

### **2.1.5 Electrocardiógrafo**

El electrocardiógrafo es un instrumento médico muy utilizado que detecta las señales eléctricas que produce el corazón asociadas a la actividad cardiaca. Es usado para diagnosticar y proponer un tratamiento para distintas enfermedades cardiacas, además de revelar cambios o tendencias en el funcionamiento del corazón. Esta herramienta es esencial en la evaluación clínica primaria en pacientes con quejas de índole cardiaco, siendo un instrumento costo-efectivo, no invasivo que conlleva muchos beneficios para el paciente (FORERO, 2018).

El electrocardiógrafo desde un punto de vista más técnico, es un aparato que detecta y registra los cambios en los voltajes generados por el corazón mediante electrodos colocados estratégicamente en el cuerpo del paciente. El gráfico obtenido muestra ondas características que reflejan la actividad eléctrica del corazón durante su ciclo, proporcionando a los médicos información crucial para evaluar la salud cardíaca, identificar irregularidades en el ritmo cardíaco y diagnosticar posibles problemas cardíacos específicos.

#### **2.1.5.1 TIPOS DE ELECTROCARDIOGRAFOS**

La categorización de los distintos tipos de electrocardiógrafos se fundamenta mayormente en la meta que el médico especialista busque alcanzar.

- **ELECTROCARDIOGRAFO MONOCANAL**

Estos dispositivos se distinguen por capturar e imprimir los datos sobre la actividad eléctrica del corazón en una única dirección de registro, requiriendo que los profesionales médicos ordenen manualmente las 12 direcciones para su análisis. A pesar de esto, la principal ventaja de este electrocardiógrafo es su peso ligero y su operación notablemente sencilla.



**Ilustración 2-2:** Electrocardiógrafo mono canal

Fuente: (KALSTEIN, 2022)

#### ■ ELECTROCARDIOGRAFO MULTICANAL

Este equipo se distingue por su capacidad de tener 3, 6 o 12 canales, capturando cada una de las 12 direcciones en un intervalo de 2,5 segundos. Algunos modelos ofrecen opciones para seleccionar distintos formatos, mientras que otros incluyen una pantalla de visualización en tiempo real que permite al usuario ver el análisis antes de imprimirlo, lo que simplifica y mejora la precisión del diagnóstico del paciente.

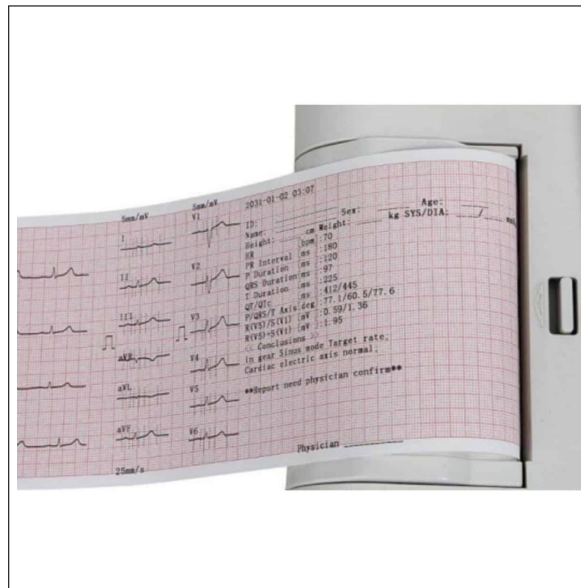


**Ilustración 2-3:** Electrocardiógrafo multi canal

Fuente: (ELECTROCARDIÓGRAFOS, 2022)

## ■ MULTICANAL CON IMPRESIÓN

Estos tipos de dispositivos operan de manera similar al multicanal, con la distinción de incluir una computadora equipada con algoritmos de reconocimiento para detectar patrones de señales de ECG, ya sean normales o anormales. Esta función proporciona un informe sugestivo al usuario, quien deberá corroborar, complementar y analizar en detalle para formular un diagnóstico preciso para el paciente.



**Ilustración 2-4:** Electrocardiógrafo multi canal con impresión

Fuente: (ELECTROCARDIÓGRAFOS, 2022)

## 2.2 Sensores para detectar frecuencia cardiaca

La información sobre la frecuencia cardíaca puede resultar valiosa al planificar un programa de ejercicios, analizar la actividad física o incluso evaluar niveles de ansiedad. No obstante, medir con precisión la frecuencia cardíaca puede presentar desafíos. Por suerte, existen algunos sensores que ofrecen una solución a esta dificultad.

### 2.2.1 Sensor Pulso

Sensor de pulso fotoeléctrico, para utilizar el sensor solo es necesario alimentarlo con un voltaje entre 3V a 5V DC y conectar la salida Analógica a la entrada analógica (ADC) de un microcontrolador (AV ELECTRONICS, 2023)



**Ilustración 2-5:** Sensor de pulso

Fuente: (AV ELECTRONICS, 2023)

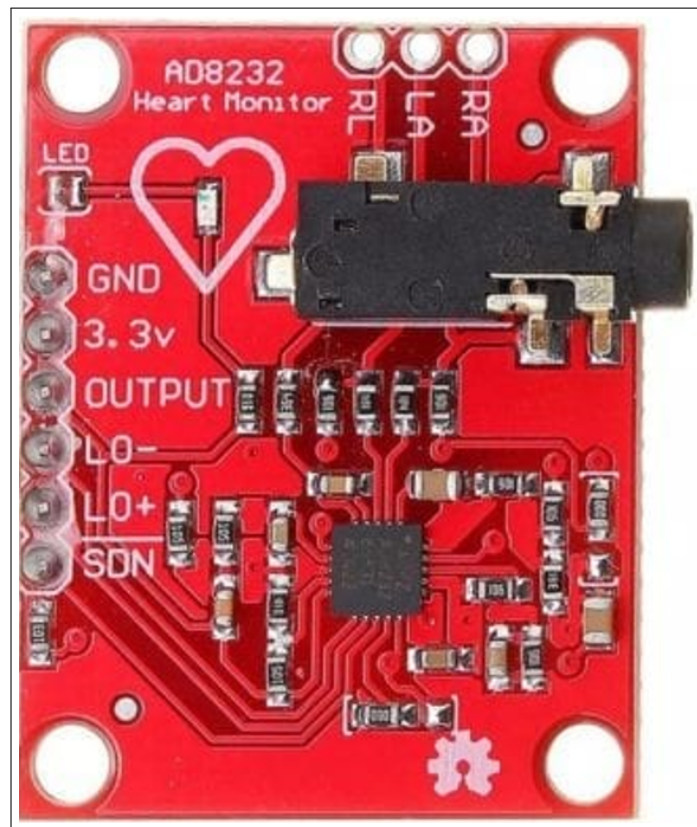
#### **Características:**

- Voltaje de Operación: 3.0V – 5.5V DC.
- Consumo corriente: 20mA máx.
- Sensor: APDS-9008.
- Opamp: MCP6001.
- Longitud de cable: 20cm.
- Cables: GND, VCC, Señal.

El sensor de pulso es un sensor de frecuencia cardíaca que solo basta conectarlo para usarlo (plug-and-play) para Arduino. Puede ser usado por estudiantes, artistas, deportistas, fabricantes, desarrolladores de juegos y aplicaciones móviles que quieran incorporar fácilmente los datos de la frecuencia cardíaca en vivo en sus proyectos (AV ELECTRONICS, 2023).

### 2.2.2 Sensor de frecuencia cardiaca ECG AD8232

AD8232 Pulso Cardíaco es una tarjeta de bajo costo usado para medir la actividad eléctrica del corazón. Esta actividad eléctrica puede ser registrada como un ECG o una salida como una lectura analógica. Los electrocardiogramas pueden ser extremadamente ruidosos, por lo que el AD8232 actúa como un amplificador operacional que ayuda a obtener una señal clara de los intervalos PR y QT fácilmente (NOVATRONIC, 2023). Los valores PR y QT son mediciones que ayudan a la evaluación de la función eléctrica del corazón, el valor PR que representa el tiempo que se lleva la propagación del impulso eléctrico desde el nodo sinusal a través de las aurículas hasta los ventrículos, varía entre 0.12 a 0.20 segundos, valores mayores indica una obstrucción entre las aurículas y los ventrículos, y el valor QT representa la duración de la despolarización y repolarización ventricular, la cual depende de la frecuencia cardiaca, es decir mayor frecuencia, el QT tiende a acortarse, y a menor frecuencia tiende a alargarse, dando este último posibles arritmias.



**Ilustración 2-6:** ECG AD8232

Fuente: (NOVATRONIC, 2023)

#### Características:

- Modelo: Modulo AD8232 ECG Pulso Cardíaco

- Dimensiones: 28mm x 35mm
- Voltaje de operación: 3.3 V
- Bajo consumo de corriente: 170  $\mu A$
- Salida de tipo analógica
- Rechazo de ruido a 60Hz: 80dB
- Configuraciones: 2 o 3 electrodos
- Ganancia elevada (G=100), con bloqueo de corriente DC
- Electrodo: 3
- Salida analógica
- Jack de 3.5 mm para la conexión de los cables para electrodos
- Led de encendido: No
- Led Indicador: Se prenderá conforme al ritmo cardíaco
- Rango de temperatura nominal: 0-70 °C
- Rango de temperatura de trabajo: -40-85 °C
- Modo de configuración ECG o electrocardiograma y la salida como lectura analógica.
- Obtiene señales de los intervalos PR Y QT

**Tabla 2-1:** Tabla comparativa entre sensores

	Sensor Pulso	ECG AD8232
Voltaje de Operación	3.0V – 5.5V DC	3.3 V DC
Consumo corriente	20mA	170 uA
Sensor	APDS-9008.	AD8232
Salida	Analógica	Analógica
Rango de temperatura de trabajo	-	-40-85 grados
Configuraciones	Pulso	2 o 3 electrodos

Realizado por: Guillermo P., 2024



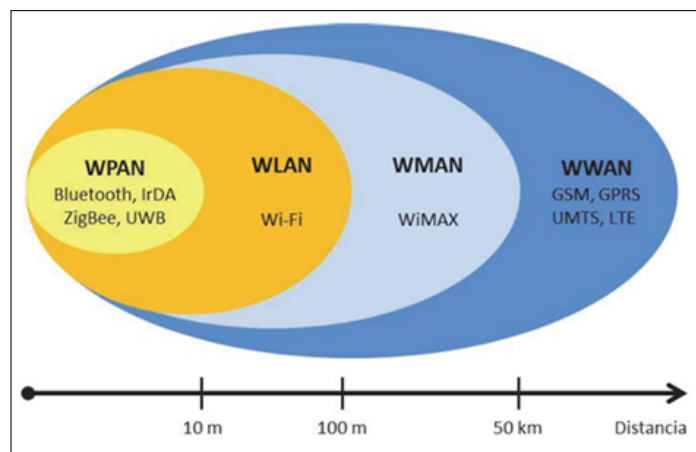
## 2.3 Redes inalámbricas

Una red inalámbrica establece conexiones entre computadoras prescindiendo del uso de cables físicos. Los dispositivos se valen de la comunicación por radio para intercambiar datos entre sí. Pueden establecer comunicación directa con otras computadoras inalámbricas o conectarse a una red preexistente a través de un punto de acceso inalámbrico (AP). Al configurar el adaptador de red inalámbrica, se elige el modo de operación que corresponde al tipo de red inalámbrica deseada.

### Características:

- No se necesitan cables físicos para establecer la conexión en la red.
- La movilidad se permite dentro del área de alcance.
- Se disminuyen los gastos de mantenimiento asociados con redes cableadas.
- Es sencillo conectar dispositivos que posean capacidades inalámbricas.

Las redes sin cables se pueden dividir en cuatro conjuntos específicos según la aplicación y el alcance de la señal, las de área personal (WPAN), las de área local (WLAN), las de área metropolitana (WMAN) y las de área amplia (WWAN). La Ilustración 2-7 muestra estas cuatro categorías.



**Ilustración 2-7:** Clasificación de las redes inalámbricas

Fuente: (SALAZAR, 2010)

Las redes inalámbricas también pueden dividirse en dos grandes grupos: corto y largo alcance. Las de corto alcance se limitan a áreas específicas, como las redes de área local

(LAN) presentes en entornos como edificios corporativos, campus escolares y universitarios, fábricas o viviendas, así como las redes de área personal (PAN) donde los dispositivos necesitan estar cercanos para comunicarse. Estas redes generalmente utilizan un espectro sin licencia reservado para usos industriales, científicos y médicos (banda ISM). Las frecuencias disponibles varían según el país, aunque las más comunes, como las de 2,4 GHz y 5 GHz, están disponibles en la mayoría de los lugares del mundo.

### **2.3.1 *Arquitectura de redes inalámbricas***

Hay dos formas de establecer la estructura de una red inalámbrica: ad hoc e infraestructura. En el modo ad hoc, los dispositivos se comunican directamente entre sí, punto a punto, mientras que en el modo infraestructura, la comunicación se realiza a través de un punto de acceso que actúa como puente hacia otras redes.

#### *2.3.1.1 Modo Ad hoc*

Cuando "se utiliza el modo ad hoc, todos los dispositivos de la red inalámbrica se comunican directamente entre sí, de igual a igual"(Salazar, SF). La red carece de una disposición estructurada o puntos estáticos, sin necesidad de un punto de acceso para la interacción entre dispositivos. El modo ad hoc es ideal para un grupo reducido de dispositivos físicamente cercanos entre sí. Sin embargo, el desempeño de la red se ve afectado a medida que aumenta el número de dispositivos.

El modo ad hoc presenta una limitación adicional: sin la presencia de pasarelas especiales, estas redes no pueden enlazarse con una red de área local cableada, lo que les impide acceder a Internet. A pesar de esto, en entornos reducidos, el modo ad hoc funciona de manera eficiente y representa la forma más sencilla y económica de establecer una red inalámbrica.

#### *2.3.1.2 Modo Infraestructura*

En esta configuración, todos los dispositivos se vinculan a la red inalámbrica a través de un punto de acceso (AP). Estos puntos de acceso suelen ser routers o switches que facilitan la transmisión de datos desde la red inalámbrica hacia una Ethernet con cable, actuando como un enlace entre la LAN cableada y los dispositivos sin cables. La conexión de varios puntos de acceso a través de una red troncal Ethernet por cable puede extender aún más la cobertura de la red inalámbrica permitiendo que un dispositivo móvil se salga del rango de cobertura de un punto de acceso y entre en el rango de otro(SALAZAR, SF).

La infraestructura brinda mayor seguridad, gestión sencilla y una notable escalabilidad y estabilidad. A pesar de estos beneficios, implica un costo extra por la instalación de puntos de acceso.

### **2.3.2 Bluetooth**

El término Bluetooth hace referencia a una tecnología de red desarrollada por el grupo de trabajo IEEE 802.15.1 del Instituto de Ingenieros Eléctricos y Electrónicos de EE. UU., adoptada como estándar industrial para conexiones inalámbricas. Esta tecnología permite la transferencia de voz y datos entre dos dispositivos digitales, ya sea sin conexión o mediante una conexión establecida. Su objetivo principal es sustituir las conexiones por cable, especialmente beneficioso para dispositivos móviles como teléfonos inteligentes o tabletas, volviendo obsoletas las conexiones cableadas entre estos dispositivos.

Bluetooth es el fruto de la colaboración entre diversas entidades. Este sistema inalámbrico tiene sus raíces principalmente en el esfuerzo de los profesores universitarios Japp Haartsen y Sven Mattisson, de nacionalidad holandesa y sueca, quienes estaban asociados con la empresa de comunicaciones móviles e internet Ericsson. Además, se debe reconocer la contribución significativa de empresas tecnológicas como Intel y Nokia. Para comprender más sobre su trasfondo tecnológico, echemos un vistazo a sus antecedentes.

La frecuencia dedicada a Bluetooth es una banda ISM sin licencia entre los 2,402 GHz y los 2,480 GHz. Los dispositivos compatibles que cumplen los estándares del Bluetooth SIG pueden, como dispositivos de corto alcance o Short Range Devices (SRD), transmitir por este rango de frecuencias en todo el mundo y sin licencia. Para poder identificarlo sin ningún género de dudas, cada aparato está provisto de una dirección MAC de 48 bits individual (IONOS, 2022). Una piconet, una red Bluetooth, puede consistir en un máximo de ocho dispositivos activos. Además, teóricamente, esta red puede mantener hasta 200 dispositivos adicionales en modo de espera o ahorro de energía dentro de la red y activarse según sea necesario. Aunque un dispositivo Bluetooth puede ser esclavo en diversas piconets, solo puede ser maestro en una. Cuando hasta diez piconets se interconectan, forman lo que se conoce como scatternet. Todos los dispositivos en esta red pueden comunicarse entre sí, aunque la velocidad de transferencia de datos se ve afectada.

#### **2.3.2.1 Clases de Bluetooth**

Los dispositivos Bluetooth se clasifican en diferentes clases dependiendo de su potencia de transmisión, y cobertura efectiva. En el caso de las clases, el límite no lo marca del

dispositivo menos potente, sino el más potente (FERNANDEZ, 2020). Por ejemplo, Si se posee un dispositivo de Clase 2, con un alcance de solo 20 metros, y el otro dispositivo es de Clase 1, con un alcance de 100 metros, la comunicación entre ambos puede extenderse más allá de los 20 metros inicialmente limitados por el dispositivo de Clase 2.

- Clase 1: Con un alcance de hasta 100 metros y un consumo promedio de potencia de 100 mW.
- Clase 2: Con un alcance de hasta 20 metros y un consumo promedio de potencia de 2,5 mW.
- Clase 3: Con un alcance de hasta 1 metro y un consumo promedio de potencia de 1 mW.
- Clase 4: Con un alcance de hasta 0,5 metros y un consumo promedio de potencia de 0,5 mW.

#### 2.3.2.2 Versiones de Bluetooth

"Las diferentes versiones que han ido apareciendo a lo largo de los años marcan sus características y lo que son capaces de hacer los dispositivos"(FERNANDEZ, 2020).

- **Bluetooth 1.0:** Lanzada en 1999, fue la primera versión de Bluetooth, pero sufrió de numerosos problemas de conectividad y seguridad debido a su naturaleza inicial. En la actualidad, está obsoleta.
- **Bluetooth 1.1:** Esta actualización, lanzada en 2002, representó una versión más estable y comercial de Bluetooth 1.0, ofreciendo una velocidad de transmisión de alrededor de 721 kbps.
- **Bluetooth 1.2:** Lanzada en 2003 como la segunda actualización de Bluetooth 1.0, esta versión abordó problemas de interferencia presentes en la versión 1.1 para mejorar aún más la tecnología.
- **Bluetooth 2.0:** Lanzada en 2004, esta versión, compatible con la 1.2, introdujo la Enhanced Data Rate (EDR o tasa de datos mejorada), permitiendo una velocidad de transmisión de más de 2 Mb/s.
- **Bluetooth 2.1:** Lanzada en 2007, mantuvo la velocidad de transmisión de su predecesora pero incorporó una característica revolucionaria: la capacidad de un dispositivo para agregarse a otro y conectarse automáticamente sin necesidad de PIN.

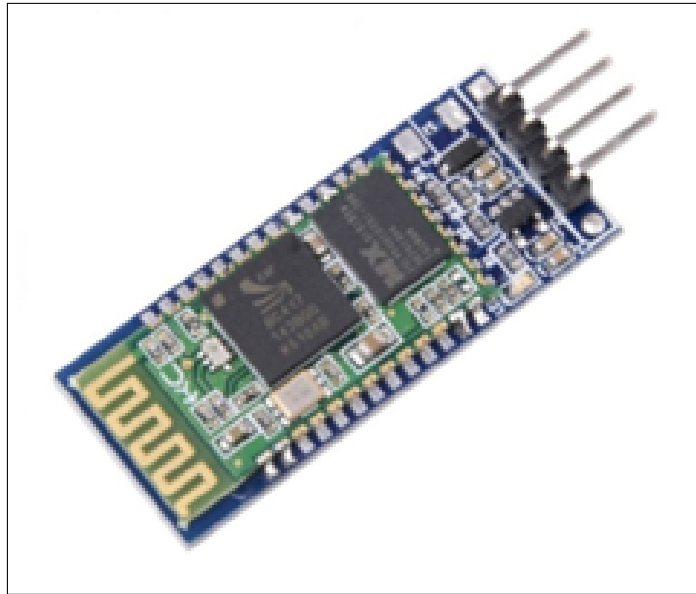
- **Bluetooth 3.0:** Lanzada en 2009, introdujo el término "HS"(High Speed o gran velocidad) y mejoró significativamente la tasa de transmisión, alcanzando hasta 24 Mbps.
- **Bluetooth 4.0:** Lanzada en 2010, esta versión comenzó a abordar el problema del consumo excesivo de energía mediante Bluetooth Low Energy (BLE). Mantuvo una tasa de transferencia de 24 Mb/s y se implementó en dispositivos más pequeños y menos potentes.
- **Bluetooth 4.1:** Lanzada en 2013, se enfocó en la conexión de dispositivos pequeños sin intermediarios, especialmente para el Internet de las cosas (IoT).
- **Bluetooth 4.2:** Lanzada en 2014, implementó el protocolo IPv6 para permitir la conexión directa a través de Internet.
- **Bluetooth 5.0:** Lanzada a mediados de 2016, esta versión representó una evolución significativa, duplicando la tasa de transferencia, cuadruplicando el alcance y manteniendo un bajo consumo de energía, ideal para el IoT. Ofreció velocidades de hasta 50 Mb/s y un alcance de hasta 240 metros.
- **Bluetooth 5.1:** Lanzada en 2019, se enfocó en la localización, permitiendo a los dispositivos conocer la ubicación de otros con una precisión de centímetros y detectar la dirección de una señal buscada.
- **Bluetooth LE Audio:** Un nuevo estándar lanzado a principios de 2020, mejora la calidad de audio y la eficiencia energética en dispositivos de audio al utilizar un nuevo códec, el Low Complexity Communication Codec (LC3). También permite la transferencia de audio a varios dispositivos simultáneamente.

### 2.3.2.3 Módulos bluetooth

Los módulos Bluetooth son dispositivos compactos ideales para proyectos que requieren conexión inalámbrica. Tienen la capacidad de funcionar como Maestros o Esclavos. La distinción radica en que un dispositivo Bluetooth Esclavo solo puede vincularse a un Maestro específico y no a otros, mientras que un Maestro Bluetooth puede conectarse con múltiples Esclavos o permitir que estos se conecten, facilitando la transmisión y recepción de información de todos ellos, gestionando las transferencias de datos (limitado a un máximo de 7 Esclavos).

#### 2.3.2.4 Módulo HC-06

El dispositivo Bluetooth HC-06 que se muestra en la Ilustración 2-8 es capaz de establecer conexiones inalámbricas utilizando el protocolo Bluetooth. Puede funcionar como esclavo o maestro, permitiendo tanto escuchar como generar solicitudes de conexión. Cuando un dispositivo se conecta, este módulo transfiere datos entre ese dispositivo y un microcontrolador en ambas direcciones.



**Ilustración 2-8:** Modulo HC-06

Fuente: (MEGATRONICA, SF)

El módulo Bluetooth HC-06 cuenta con 4 pines y opera exclusivamente como esclavo. Ofrece un conjunto limitado de comandos AT para facilitar una conexión simple y sin complicaciones a través de una interfaz serie. Además, posibilita la transmisión de voz y datos mediante una red inalámbrica conocida como WPAN.

Los módulos HC-06 se ensamblan sobre una interfaz que incluye cuatro pines: Vcc, Gnd, Txd y Rxd. Además, disponen de un LED que indica el estado de la conexión Bluetooth: el parpadeo señala que no está emparejado, mientras que una luz continua indica que la conexión Bluetooth está establecida. Estos módulos cumplen con las especificaciones del estándar Bluetooth 2.0 a 2.4 GHz y son perfectamente compatibles con dispositivos como Arduino y PIC, así como con teléfonos Android, pero no son compatibles con iPhones.

#### 2.3.2.5 Funciones

- Sustituye un enlace serial por uno inalámbrico sin cambios perceptibles.

- Incluye el perfil Bluetooth "Puerto Serie Inalámbrico".
- Puede ajustarse para operar como Maestro o Esclavo.
- Excelente para supervisar proyectos de Arduino, PIC, Atmel, entre otros, de manera inalámbrica desde una computadora.

## 2.4 Comunicaciones móviles

La telefonía es el servicio principal de los sistemas de comunicación móvil, presente en todos ellos. Aunque inicialmente era analógica, actualmente ha evolucionado hacia lo digital. Esta forma digital de telefonía móvil, llamada también celular en algunos lugares, se fundamenta en el concepto de celdas o células.

La conexión se desarrolla en zonas geográficas amplias, abarcando la mayoría de la población en un país. Dividir el área de servicio en células con estaciones base dedicadas permite ofrecer un servicio de alto rendimiento a numerosos usuarios. Cuando un usuario se mueve de una célula a otra, se lleva a cabo un proceso complejo de cambio de célula, conocido como "handover", que asegura la continuidad de la llamada. Un sistema de telefonía móvil celular está compuesto por muchos elementos y dispositivos. Entre los más importantes figuran las estaciones o terminales móviles, las estaciones base y los centros de conmutación móvil (MSC según las siglas inglesas). El MSC es el encargado de conectar los móviles a la red de telefonía fija. Cada móvil se conecta vía radio con una de las estaciones base del sistema. Durante la llamada se pueden realizar todos los procedimientos de "handover" necesarios"(GARCIA et al., 2014). .

"La comunicación entre la estación base y los móviles se define por un estándar denominado "Common air interface" (CAI) que especifica cuatro canales diferentes. Los canales usados para la transmisión de voz desde la estación base a los móviles se llaman "forward voice channels" (FVC) y los canales utilizados en la transmisión de voz en sentido contrario se denominan "reverse voice channels" (RVC). Los otros dos canales son canales de control y se denominan "forward control channels" (FCC) y "reverse control channels" (RCC). Estos canales también se llaman canales de establecimiento ("set up channels") ya que son utilizados en el establecimiento de la llamada. Los canales de control transmiten y reciben datos que contienen las peticiones de inicialización y de servicio"( GARCIA et al., 2014). En las últimas dos décadas, los sistemas de telefonía móvil han experimentado un crecimiento significativo, destacándose el estándar europeo GSM como uno de los más exitosos. Este sistema, digital y celular, ha sido uno de los más prominentes. Asimismo, en Estados Unidos se implementó el sistema D-AMPS, otro ejemplo de este tipo de tecnología

celular.

#### 2.4.0.1 Cuarta generación 4G

4G es la abreviatura que representa la aspiración de ser la cuarta generación de tecnologías de telefonía móvil. Se fundamenta completamente en el protocolo IP, convirtiéndose en un sistema de sistemas y una red compuesta por varias redes. No se trata de una tecnología o estándar específico, sino de una amalgama de tecnologías y protocolos diseñados para maximizar la capacidad de procesamiento. Surge tras la convergencia entre redes cableadas e inalámbricas, así como con la integración en computadoras, dispositivos electrónicos y tecnologías de la información. Esto posibilita alcanzar velocidades de acceso entre 100 Mbps en movimiento y 1 Gbps en reposo, manteniendo la seguridad en las comunicaciones punto a punto y permitiendo la prestación de servicios diversos en cualquier momento, con costes mínimos.

LTE utiliza la banda de frecuencia de 700 MHz, aprovechando su disponibilidad tras la transición de la televisión analógica, para garantizar una mejor cobertura y penetración en estructuras, aspecto crucial para las operadoras que ofrecen este servicio. En cuanto al funcionamiento de la tecnología LTE, se distingue entre su operatividad en el canal de descarga de datos y en el canal de carga de datos:

- En el canal de descarga, LTE emplea una técnica de modulación llamada OFDMA (Acceso Múltiple por División Ortogonal de Frecuencia). Las subportadoras se modulan utilizando símbolos en un rango que incluye QPSK, 16QAM o 64QAM. Esta técnica ofrece una alta resistencia a los efectos del multipath, siendo ideal para implementaciones como MIMO o SFN.
- Para la transmisión de datos en el canal de carga, LTE utiliza una técnica llamada SC-FDMA (Acceso Múltiple de División Simple de Portadora) para simplificar el diseño y reducir tanto los picos de la tasa de bits como el consumo de energía.

Uno de los principales propósitos de esta tecnología es ofrecer velocidades de descarga y carga de archivos significativamente más altas que las logradas con las tecnologías vigentes. En términos formales, los principales objetivos del LTE son los siguientes:

- Alcanzar velocidades de transmisión más altas, llegando hasta 100 Mbps en descarga y 50 Mbps en carga.



- Reducir la latencia a menos de 10 milisegundos y lograr tiempos de establecimiento de comunicación iniciales inferiores a 100 ms.
- Lograr una eficiencia espectral tres veces superior a la del HSPA.
- Mejorar la prestación de servicios de difusión para permitir la transmisión y recepción de servicios de radiodifusión y televisión móvil en tiempo real, ofreciendo una calidad aceptable para todos los usuarios.
- Garantizar flexibilidad en el espectro, permitiendo ajustar los anchos de banda según el servicio y utilizar diversas bandas de frecuencia de acuerdo con las características de cada área geográfica.

Para alcanzar la consecución de estos objetivos se están planteando arquitecturas de red con tendencia a simplificar lo máximo posible la jerarquía, hablando de estructuras planas. En éstas, la radio cobra un gran protagonismo, ya que debe asumir funciones que actualmente se hallan distribuidas en otras plataformas (PEREZ, 2010).

#### 2.4.0.2 Quinta generación 5G

La designación 5G hace referencia a la quinta generación de redes móviles. Se distingue de las anteriores, como la antigua red 1G, que estaba limitada únicamente a la comunicación por voz en los primeros teléfonos móviles. "La evolución tecnológica comenzó con la introducción de los SMS en la red 2G, transformando gradualmente nuestros smartphones en herramientas de comunicación más avanzadas. La incorporación de la conexión a Internet (3G) fue el siguiente paso, seguido por la llegada de la banda ancha (4G), lo que permitió la reproducción de videos en tiempo real (streaming) y la realidad aumentada, aspectos que hoy son habituales pero que hace algunos años eran completamente inimaginables. El avance más significativo vendrá de la mano de la velocidad. El 5G permitirá navegar hasta a 10 GBps (gigabytes por segundo), 10 veces más rápido que las principales ofertas de fibra óptica del mercado. A ese ritmo se podrá, por ejemplo, descargar una película completa en cuestión de segundos"(NATIONAL GEOGRAPHIC, 2022).

Además, se espera un avance significativo en la latencia, es decir, el tiempo de respuesta de la red. Según los operadores, esta latencia podría reducirse a 5 milisegundos, un intervalo casi imperceptible para los humanos. Esto nos posibilitará establecer conexiones prácticamente en tiempo real. Este aspecto cobra especial relevancia, por ejemplo, al minimizar el tiempo de respuesta de un vehículo autónomo, lo que contribuirá a mejorar la seguridad tanto de sus ocupantes como de los peatones cercanos.

Esta nueva tecnología permitirá, entre otras cosas, incrementar de forma considerable la cantidad de dispositivos conectados. Vehículos, robots industriales, elementos del mobiliario urbano (como badenes, señalización vial, paradas de autobús) o cualquier dispositivo electrónico en nuestros hogares (desde alarmas hasta electrodomésticos como lavadoras, neveras o robots aspiradores) podrán establecer conexiones y compartir información en tiempo real.

## **2.5 Tarjetas de desarrollo**

Una tarjeta de desarrollo es una herramienta "que permite a un usuario la realización de diseño de prototipos tecnológicos, cuentan con una unidad principal de procesamiento de información, así como la integración, dentro del mismo dispositivo, de los elementos necesario para realizar diferentes y variados proyectos de desarrollo y de investigación"(GISI, SF).

Los elementos principales presentes en estas placas incluyen la memoria, el bus de comunicación y varios puertos de entrada/salida como USB, Ethernet, pines digitales/análogos, con la opción de conectividad Wi-Fi en algunos casos. Algunas también están diseñadas con características especiales para integrarse en sistemas de comunicaciones móviles e Industria 4.0.

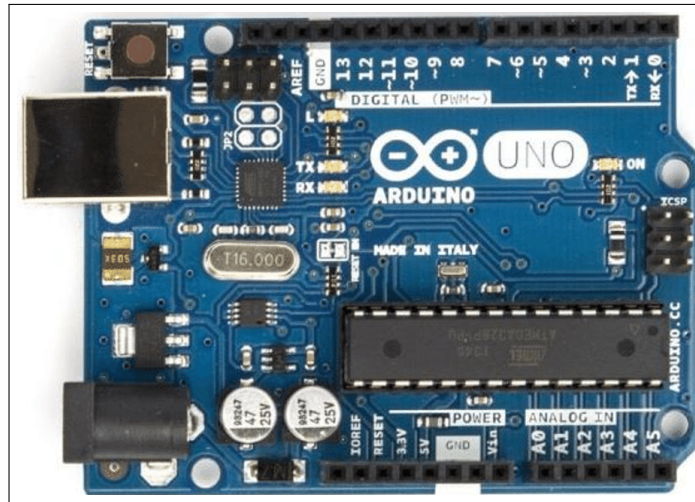
Recientemente, ha existido un aumento en la disponibilidad de tarjetas de desarrollo en el mercado, cada una con distintas cualidades y atributos. Basta con hacer una rápida búsqueda en tiendas minoristas como Amazon para acceder a una amplia gama de opciones. Dada esta diversidad, resulta difícil recopilar información sobre todas ellas en esta publicación. No obstante, se mencionan algunas de las placas más populares.

### **2.5.1 Arduino**

Probablemente la tarjeta más reconocida a nivel global por su objetivo de simplificar la programación y el diseño de prototipos se el arduino uno que se puede visualizar en la Ilustración 3-5. Su característica principal radica en que, aunque existe el proyecto oficial Arduino con sus placas originales disponibles en su sitio web, esta tarjeta se desarrolla bajo el concepto de software y hardware de código abierto, lo que permite a cualquier persona replicarla sin restricciones.

Esta placa ha ganado una gran popularidad por su facilidad de uso en proyectos académicos, técnicos y científicos. Una simple búsqueda de proyectos Arduino en cualquier

motor de búsqueda revela una amplia variedad de posibilidades para trabajar con esta plataforma. Algunos críticos de esta herramienta cuestionan su facilidad de programación. No obstante, es importante recordar que esta facilidad ha acercado a millones de estudiantes y personas en general al campo del diseño ingenieril, permitiendo la realización de proyectos de diversa complejidad según la creatividad del usuario.



**Ilustración 2-9:** Arduino UNO

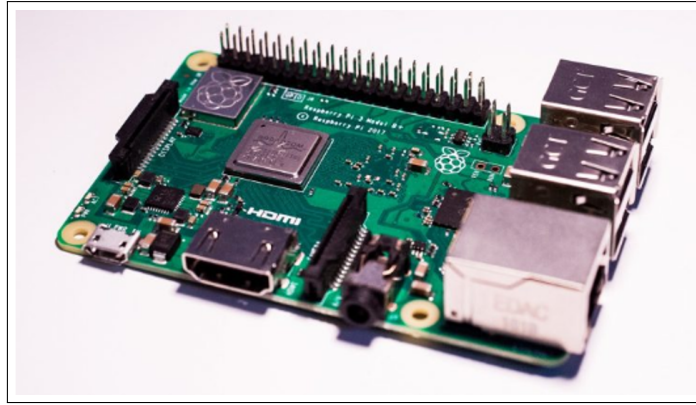
Fuente: (VARGAS, 2024)

### 2.5.2 RASPBERRY PI

Esta placa visualizada en la Ilustración 2-10, reconocida como una Computadora de Placa Única (SBC por sus siglas en inglés), ha ganado una enorme popularidad en la actualidad. Fue creada por la Raspberry Pi Foundation en el Reino Unido con el propósito inicial de acercar a las personas al mundo de la informática. Sin embargo, su uso se expandió rápidamente hacia el desarrollo tecnológico en general.

La Raspberry es esencialmente una computadora completa, equipada con todos los periféricos comunes que se encuentran en una PC doméstica. Incluye unidades de almacenamiento, la capacidad de conectarse a monitores y la posibilidad de conectar ratones y teclados. Además, es compatible con diversos sistemas operativos, siendo el más popular Raspbian, una variante de Debian.

Aunque no fue inicialmente concebida para la creación de prototipos, esta tarjeta ha ganado una enorme popularidad en ese campo, convirtiéndose en el componente central de numerosos proyectos de domótica, comunicaciones móviles e industria 4.0.



**Ilustración 2-10:** Raspberry PI

Fuente: (BENIGNO, 2020)

### 2.5.3 ESP32

El ESP32 observado en la Ilustración 2-11, es una línea de microcontroladores asequibles y de bajo consumo que incorpora Wi-Fi y Bluetooth de doble modo en un sistema en chip (SoC). Este avance resulta ventajoso para los ingenieros en automatización que prefieren evitar los aspectos más complejos de la radiofrecuencia (RF) y el diseño inalámbrico. Esta serie, que funciona como una combinación de radio Wi-Fi/Bluetooth económica, ha ganado popularidad no solo entre los entusiastas, sino también entre los desarrolladores de IoT. Su eficiencia energética, variedad de entornos de desarrollo de código abierto y bibliotecas la convierten en una elección idónea para desarrolladores de diversos ámbitos. No obstante, la serie ESP32 se presenta en numerosos módulos y placas de desarrollo, lo que puede dificultar la selección de los modelos adecuados.

El módulo ESP32 se presenta como una solución integral y certificada de Wi-Fi/Bluetooth, que no solo incluye la radio inalámbrica, sino también un procesador incorporado con interfaces para la conexión a diversos dispositivos periféricos.



**Ilustración 2-11: ESP32**  
Fuente: (BENIGNO, 2020)

#### 2.5.4 PHOTON

El Photon visto en la Ilustración 2-12, es una pequeña placa de desarrollo que incluye Wi-Fi integrado, ideal para proyectos de Internet de las cosas. Es fácil de utilizar, compatible con la programación de Arduino, potente y se conecta directamente a la nube. Se considera la versión actualizada que sustituye al Spark Core.

Las herramientas que componen el entorno del Photon, incluidas con la placa, están diseñadas para la construcción y creación de dispositivos, ya sea para ingenieros, desarrolladores, entusiastas de Arduino o emprendedores en productos de comunicaciones móviles.



**Ilustración 2-12: PHOTON**  
Fuente: (LEMUS, 2019)

## **2.6 Creación de aplicaciones móviles**

La plataforma de desarrollo de aplicaciones engloba un conjunto integral de herramientas interconectadas necesarias para crear, implementar y actualizar aplicaciones. Ofrece diversas funciones para cada etapa del proceso, abarcando diseño, programación, implementación, integración, pruebas, entre otros. Estas herramientas también facilitan actualizaciones, estrategias publicitarias y de marketing.

Para crear esas aplicaciones móviles, es esencial emplear plataformas de desarrollo de aplicaciones, las cuales se pueden clasificar en distintas categorías según su enfoque específico.

### **2.6.1 NATIVAS**

Las plataformas de desarrollo nativas son aquellas diseñadas específicamente para cada sistema operativo, como iOS, Android o Windows Phone. Esto implica el uso de un lenguaje de programación adaptado a cada SO: Objective-C para iOS, Java para Android y .Net para Windows Phone.

Estas aplicaciones tienen la ventaja de aprovechar al máximo las capacidades de los dispositivos y funcionar sin conexión a internet, lo cual marca una diferencia notable, sin embargo, las plataformas de desarrollo nativas también presentan un inconveniente: tanto el desarrollo inicial como las actualizaciones de estas aplicaciones implican costos significativos.

### **2.6.2 HIBRIDAS**

Las aplicaciones híbridas combinan características de las aplicaciones nativas y la web, ajustándose a requerimientos específicos. Se construyen usando lenguajes como JavaScript, CSS o HTML, similares a los utilizados en las aplicaciones web, lo que les permite adaptarse a diversos sistemas operativos. Asimismo, al igual que las aplicaciones nativas, ofrecen acceso completo a las capacidades de los dispositivos.

### **2.6.3 MULTIPLATAFORMA**

Una elección muy solicitada por los desarrolladores por su capacidad para disminuir costos y el tiempo necesario para su creación. Estas aplicaciones, como su nombre indica, se ajustan específicamente a las distintas plataformas de los dispositivos móviles.

El único inconveniente que presentan es que los usuarios no pueden aprovechar al máximo todas las funcionalidades de los teléfonos y tabletas mediante estas aplicaciones.

#### **2.6.4 WEB/HTML5**

Este tipo de aplicaciones son universales y se ajustan a todos los sistemas operativos, evitando así la necesidad de crear una aplicación para cada SO, como se hace con las aplicaciones nativas. También se adaptan con precisión a los navegadores móviles de los dispositivos.

Utilizando HTML5, los programadores y desarrolladores tienen la capacidad de crear aplicaciones web que los usuarios pueden utilizar desde cualquier dispositivo móvil a través del navegador web incorporado en el mismo terminal.

#### **2.6.5 APP INVENTOR**

App Inventor busca democratizar la creación de software al capacitar a los jóvenes para que dejen de ser meros usuarios de tecnología y se conviertan en creadores activos de esta, con este propósito, App Inventor se ha diseñado como un software de programación altamente visual e intuitivo que emplea un lenguaje basado en bloques.

Esta aplicación busca hacer más accesible la creación de software al empoderar a los jóvenes, transformándolos de simples consumidores pasivos de tecnología en participantes activos en su creación.



**Ilustración 2-13:** APP inventor

**Fuente:** (RPEREXP, 2022)

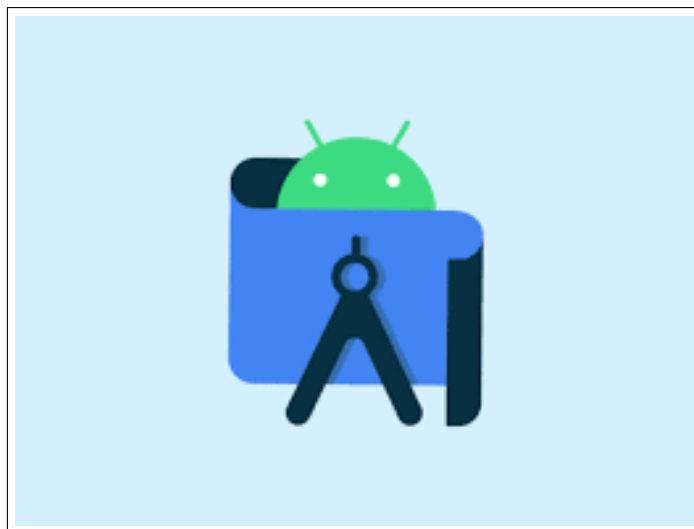
#### **2.6.6 ANDROID STUDIO**

Cuando mencionamos Android Studio, nos referimos a un entorno de desarrollo especializado para la creación de herramientas y aplicaciones destinadas a sistemas operativos

Android. Cada aplicación o herramienta diseñada para este sistema operativo tiene su propio espacio de trabajo dentro de Android Studio, ofreciendo flexibilidad en el desarrollo de funciones y características específicas para estas herramientas o aplicaciones, el icono representativo de esta plataforma se muestra en la Ilustración 2-14

Este entorno favorece la eficiencia y autonomía de las aplicaciones en desarrollo, incluso permitiendo la compatibilidad con otras plataformas o sistemas. Android Studio facilita la integración de características y funciones que se van mejorando con el tiempo, ofreciendo un proceso de refinamiento continuo para las aplicaciones.

- La finalidad de esta plataforma es brindar al usuario una experiencia de trabajo fluida y con una amplia variedad de funciones prácticas y beneficiosas.
- Este entorno posibilita el desarrollo de aplicaciones para dispositivos Android en general.
- Incluye plantillas predefinidas que agilizan la incorporación de funciones comunes de otras aplicaciones, así como la capacidad de importar códigos de muestra.
- Ofrece una mayor diversidad de herramientas de prueba con frameworks.
- Permite la edición de secciones de código y recursos de una aplicación sin necesidad de reiniciarla.



**Ilustración 2-14:** Android studio

Fuente: (SANTAELLA, 2022)



## 2.7 SERVICIOS CLOUDING

Los servicios de nube son estructuras, plataformas o sistemas informáticos que proveedores externos albergan y ofrecen a los usuarios a través de la red como se muestra en la Ilustración 2-15. Estos servicios agilizan el intercambio de datos entre los usuarios, sus dispositivos front-end, y los sistemas de los proveedores, permitiendo el desarrollo y la flexibilidad de trabajar en aplicaciones en la nube. Acceder a estos servicios solo requiere que los usuarios dispongan de una computadora, un sistema operativo y conexión a Internet.

Los servicios de computación en la nube comprenden todas las infraestructuras, plataformas, tecnologías o sistemas de software a los que los usuarios acceden mediante Internet, prescindiendo de la necesidad de descargar software adicional.



**Ilustración 2-15:** Servicios Clouding

Fuente: (RED-HAT, 2023)

### 2.7.1 FUNCIONAMIENTO DE LOS SERVICIOS DE NUBE (FIREBASE)

Al igual que otras soluciones en tecnologías de la información, los servicios de nube se basan en sistemas de hardware y software, pero se distinguen por el hecho de que los usuarios solo requieren una computadora, conexión a internet y un sistema operativo para acceder a ellos. Firebase es una herramienta desarrollada por Google, se presenta como una plataforma de computación en la nube diseñada específicamente para facilitar el proceso de desarrollo de aplicaciones, ya sea para entornos web o dispositivos móviles. Su principal ventaja radica en su capacidad para trabajar de manera eficiente en diversas plataformas, como iOS, Android y la web, lo que contribuye a agilizar significativamente el proceso de desarrollo de aplicaciones.

Las utilidades que ofrece son diversas y accesibles, lo que permite simplificar la gestión al centralizar las actividades en una única plataforma. Estas herramientas se clasifican en cuatro categorías principales: desarrollo, expansión, generación de ingresos y análisis. Resulta especialmente beneficioso para los desarrolladores, ya que les permite reducir el tiempo dedicado a tareas de backend, tanto en el proceso de desarrollo como en el mantenimiento posterior, este servicio se define como modelo de servicio BaaS

#### 2.7.1.1 *BaaS*

Conocido también como backend como servicio, representa un modelo de servicio en la nube donde los desarrolladores subcontratan todas las funciones de backend de una aplicación móvil o web. Esto les permite dedicar más tiempo a gestionar la interfaz y mejorar la experiencia del usuario con sus aplicaciones.

#### 2.7.1.2 *Realtime Database*

Una de las características más notables y fundamentales de Firebase son sus bases de datos en tiempo real, las cuales residen en la nube y adoptan un enfoque NoSQL para almacenar los datos en formato JSON. Estas bases de datos permiten alojar y acceder a los datos de la aplicación en tiempo real, garantizando su actualización constante incluso cuando el usuario no interactúa con la aplicación. Firebase automáticamente envía eventos a las aplicaciones cuando hay cambios en los datos, guardando las actualizaciones en disco. Incluso si un usuario pierde la conexión, los datos siguen siendo accesibles para otros usuarios, y los cambios realizados se sincronizan una vez que se restablece la conexión.

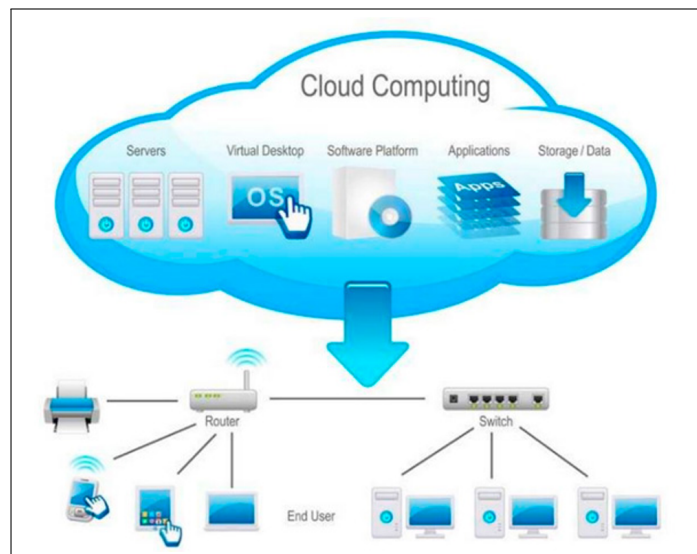
### 2.7.2 **INFRAESTRUCTURA DE LA NUBE**

Cuando los proveedores ofrecen infraestructura en la nube, desacoplan las capacidades informáticas de los componentes de hardware, como se ilustra en los siguientes ejemplos:

- La capacidad de procesamiento de las CPU.
- La memoria activa proporcionada por los chips de RAM.
- El procesamiento gráfico ofrecido por las GPU.
- El espacio de almacenamiento disponible en discos duros o centros de datos.

Usualmente, esta separación se lleva a cabo mediante la virtualización y el uso de máquinas virtuales. A través de Internet, los usuarios acceden a estos recursos informáticos, de

almacenamiento y de red como se indica en la Ilustración 2-16, formando así una infraestructura o IaaS. Este tipo de servicio ha contribuido al crecimiento del almacenamiento en la nube, donde se guarda el big data como parte de las comunicaciones móviles.



**Ilustración 2-16:** Infraestructura de nube

Fuente: (RED-HAT, 2023)

La infraestructura de la nube es un conjunto de tecnologías que posibilita almacenar, procesar y administrar datos y programas utilizando internet. Se compone de centros de datos conectados globalmente que alojan servidores, almacenamiento y redes. En esta estructura, los servidores físicos y virtuales forman la base, siendo los físicos máquinas reales y los virtuales entornos lógicos creados a partir de los físicos.

El almacenamiento en la nube ofrece variedad, desde almacenamiento de bloques hasta almacenamiento de archivos y objetos, proporcionando espacio para guardar datos de usuarios y aplicaciones. La red es esencial para la conectividad entre usuarios, servidores y servicios en la nube, empleando dispositivos como enrutadores, switches y firewalls para garantizar la seguridad en la transmisión de datos, a través de la nube, se accede de forma remota a estos recursos mediante internet. Los usuarios pueden almacenar información, ejecutar programas y usar servicios sin necesidad de tener la infraestructura física. Esto permite ajustar capacidades según necesidades específicas, brindando flexibilidad y eficiencia sin requerir grandes inversiones en hardware.

## CAPÍTULO III

### 3. MARCO METODOLÓGICO

En este capítulo se desarrollan y ejecutan cada una de las etapas para el diseño y la implementación del prototipo de un electrocardiógrafo móvil con tecnología bluetooth y la red celular, abordando el sistema y los elementos esenciales para llevar a cabo el trabajo de Integración Curricular. También se detallan las particularidades y la estructura de las fases de Hardware y Software del modelo, contemplando los diversos componentes requeridos para satisfacer las necesidades y objetivos planteados en el proyecto.

#### 3.1 Requerimientos del sistema

Para definir los requerimientos del sistema se considera toda la información recabada en el Capítulo II, considerando eficiencia, economía y factibilidad de cada todos los elementos necesarios para el desarrollo del proyecto.

##### 3.1.1 *Requerimientos del hardware del sistema*

Dentro del dispositivo físico del prototipo se encuentra los sensores y una placa de conectividad, que se diseña debe cumplir los siguientes requerimientos, los cuales influirán de forma significativa en la implementación, los cuales son:

- Ser de bajo coste, portable y fácil uso.
- Proporcionar la información en tiempo real del comportamiento eléctrico del corazón de un paciente.
- Transmitir toda la información recaba del corazón con una latencia mínima.
- Tener la capacidad de movimiento en cuanto a energía, significando que no debe depender de una red eléctrica doméstica.
- Mantener la comunicación el tiempo necesario para que se pueda hacer un monitoreo del comportamiento del corazón.

##### 3.1.2 *Requerimientos del software del sistema*

El software que se desarrollara para la implementación de prototipo está relacionado con los entornos de desarrollo para aplicaciones móviles (Android) y la página web, conside

rando la compatibilidad entre ambos, así como una serie de requerimientos necesarios para el correcto funcionamiento del prototipo, en relación a la aplicación móvil los requerimientos necesarios se describen a continuación:

- Tener una interfaz que sea amigable con el usuario.
- Ser de fácil manejo y portable para el usuario.
- Permitir la comunicación bluetooth con el módulo bluetooth del electrocardiógrafo.
- Permitir ingresar el número de cédula del paciente y visualizar los datos de la frecuencia cardíaca.
- Capacidad de crear perfiles en la nube para visualizar y guardar los datos.
- Reiniciar todo permitiendo crear nuevos registros de datos de este o de diferentes pacientes.

Y los requerimiento con respecto a la página web son:

- Fácil manejo, portable y liviano.
- Permitir visualizar el número de cédula y los datos guardados de su frecuencia cardíaca.
- Permitir eliminar los registros de los pacientes.

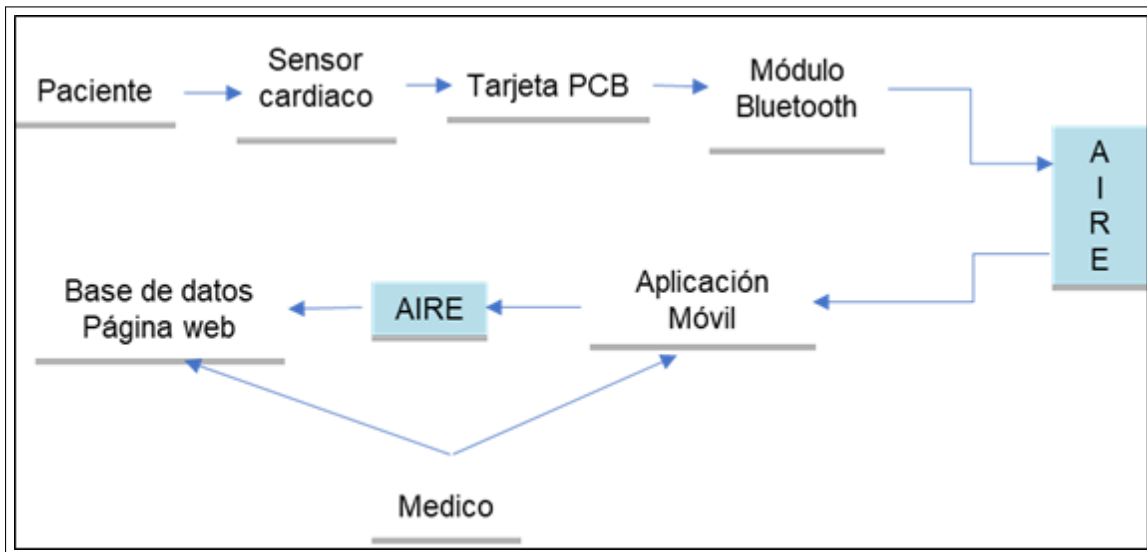
### **3.2 Descripción general del sistema**

El sistema completo del trabajo de titulación debe cumplir con distintos requerimientos que facilite a la implementación, así como futuras mejoras del mismo, las cuales se describen a continuación:

- Un sistema de bajo coste y escalable.
- El sistema en todo su conjunto debe ser de fácil uso para el usuario.
- Los datos obtenidos desde el sensor deben poder guardarse y visualizarse en tiempo real.
- El prototipo debe contar con la característica de portabilidad, es decir, debe proveerse de energía desde una batería que además pueda ser recargable o reemplazable.

### 3.3 Concepción del sistema

El presente trabajo de integración curricular está enfocado en el desarrollo de un prototipo de un electrocardiógrafo de bajo costo que tenga la capacidad de conectarse a un dispositivo móvil utilizando modelos bluetooth para poder enviar la información recaba por el electrocardiógrafo hacia el dispositivo móvil y a su vez se genere una base de datos en la nube con el número de cedula del paciente para poder observar los datos de frecuencia cardiaca obtenidos con el electrocardiógrafo, proceso que se visualiza en la Ilustración 3-1



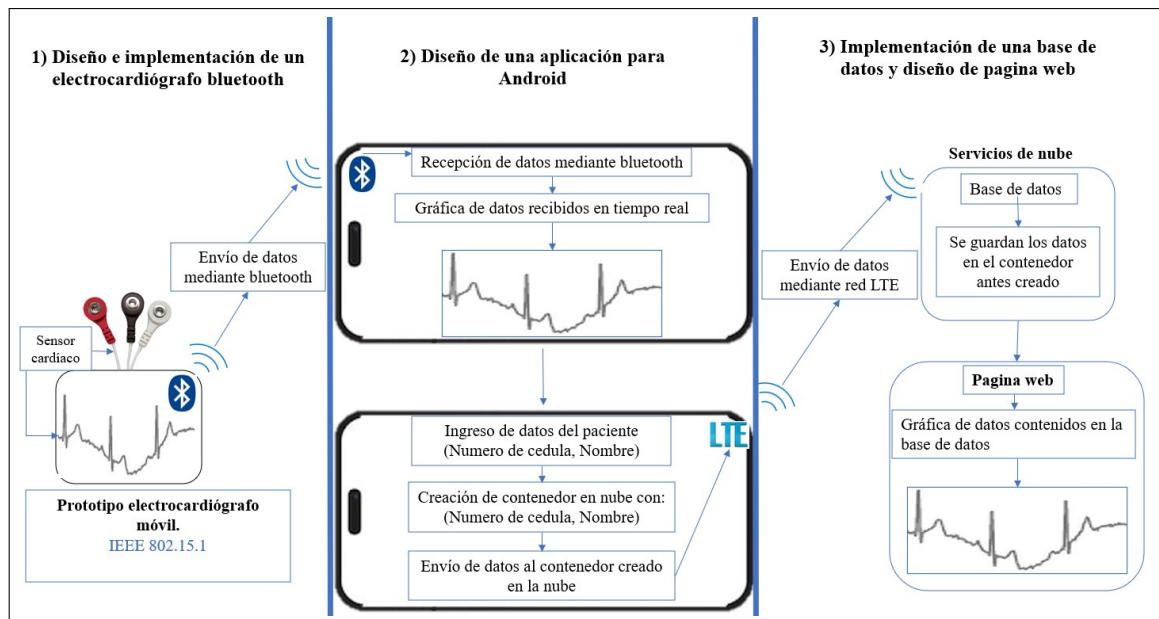
**Ilustración 3-1:** Diagrama de bloques del sistema.

Realizado por: Guillermo P., 2024

Para el funcionamiento del prototipo el paciente tiene que colocarse los electrodos del luego del después, móvil que verificargráfica la frecuencia cardiaca, después, se tiene que agregar el número de cedula del paciente para poder transmitir todos los datos a la nube.

#### 3.3.1 Metodología para el desarrollo del prototipo (Mockup)

Segun Salazar "La práctica de construir mockups para prototipar interfaces de usuarios y otros aspectos de las aplicaciones se ha vuelto cada vez más frecuentes en los procesos de desarrollos modernos"(SALAZAR, 2014). En la ilustración 3-2 se describe la metodología muckup, la cual está definida en 3 etapas principal: diseño e implementación de un electrocardiógrafo bluetooth, diseño de una aplicación para Android e implementación de una base de datos y diseño de página web, se define cada uno de las etapas con subprocesos los cuales tienen un orden cronológico de desarrollo y de funcionamiento del prototipo en conjunto con el sistema de comunicación.



**Ilustración 3-2:** Metodología Mockup.

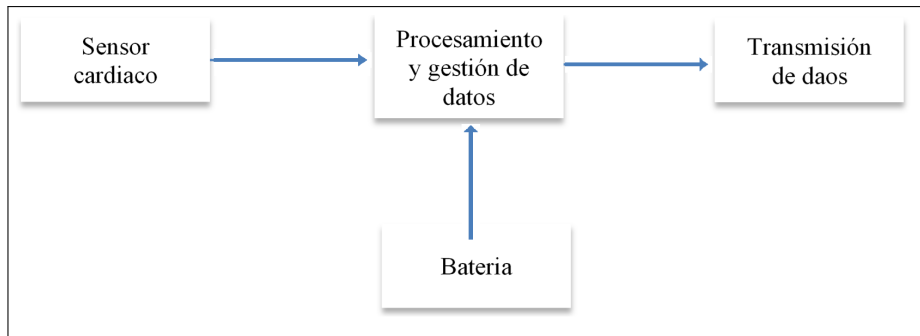
Realizado por: Guillermo P., 2024

### 3.4 Arquitectura del sistema

El prototipo de electrocardiograma móvil está conformado por dos partes, una de ellas es el nodo lector transmisor y el nodo recolector, los cuales se representan por dos diagramas de bloques, cada uno de ellos se los representa a continuación.

#### 3.4.1 *Nodo transmisor*

En la ilustración 3-3 se muestra el diagrama de bloques del nodo transmisor lector, el cual está conformado por 4 bloques iniciando por el sensor cardiaco el cual tiene la función de recabar toda la información de índole eléctrica que produce el corazón debido a su funcionamiento, esta información es monitoreada de forma constante por periodos de tiempo de esta misma forma el bloque de "Procesamiento y gestion de datos" representa una placa basada en Arduino la cual es la encargada de gestionar los tiempos de muestreo y procesamiento de la señal recabada, esta cuenta con un módulo bluetooth. Para que todo el sistema funcione se alimenta de una batería de 3.7 voltios, se utiliza una batería recargable con puerto micro USB, la cual tiene una autonomía de 20 horas, como también puede estar conectado a una laptop o power band.



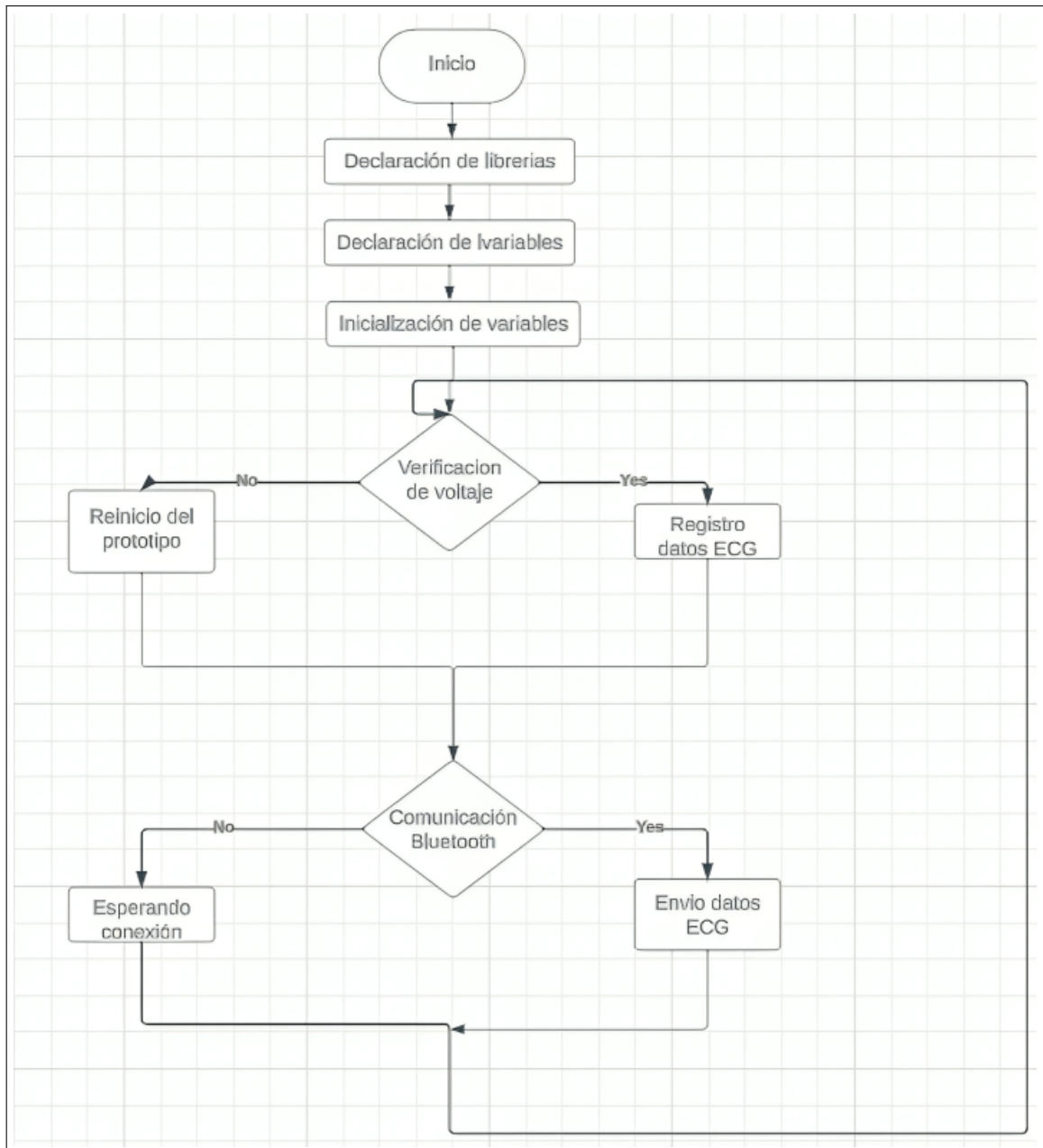
**Ilustración 3-3:** Diagrama de bloques del sistema, nodo transmisor lector

Realizado por: Guillermo P., 2024

Basado en el diagrama del nodo lector, se realiza el análisis del diagrama de flujo para la implementación del código para conocer las acciones necesarias para el correcto funcionamiento de este nodo.

Mencionando la fuente de alimentación se utiliza una batería recargable con puerto micro USB, el cual tiene una autonomía significativa, como también puede estar conectado a una laptop o power band.





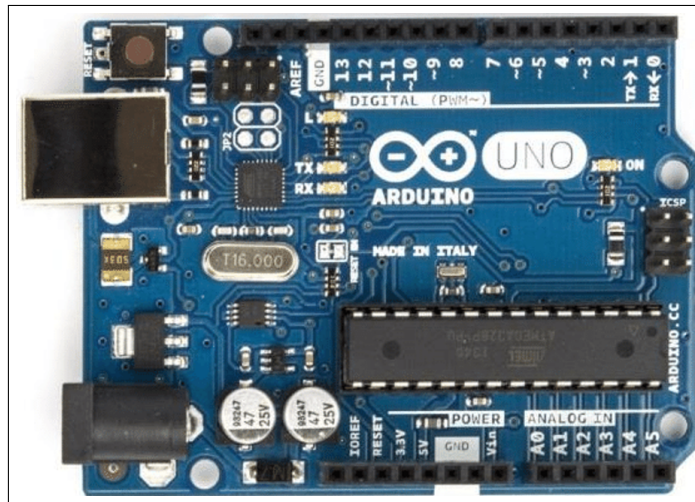
**Ilustración 3-4:** Diagrama de flujo, nodo transmisor lector

Realizado por: Guillermo P., 2024

En el desarrollo del nodo transmisor lector se define una serie de componentes necesarios para su correcto funcionamiento, entre los componentes se encuentran sensores, batería, tarjeta de desarrollo, y módulos de comunicación, los cuales se describen a continuación, las principales características, códigos y conexiones.

### 3.4.1.1 Tarjeta de desarrollo (Arduino UNO)

La tarjeta de desarrollo que se utiliza para la elaboración del prototipo, es el Arduino UNO, el cual se visualiza en la Ilustración 3-5, siendo un microcontrolador de código abierto que utiliza el microcontrolador ATmega 328P, cuenta con un conjunto de pines, 14 pines digitales y 6 pines analógicos, un oscilador de cristal de 16 MHz, para la programación es a través de Arduino IDE, a través de un cable USB.



**Ilustración 3-5:** Arduino UNO

Fuente: (VARGAS, 2024)

Existen pines de alimentación necesarios para el funcionamiento de la tarjeta de desarrollo, los cuales son seleccionados dependiendo del objetivo del proyecto, estos pines son descritos en la Tabla 3-1.

Entiendo. Si deseas mantener el texto Realizado por.<sup>a</sup>alineado con la tabla pero fuera de ella, puedes usar el entorno minipage dentro de la tabla, pero esta vez con la opción b para alinearlo a la parte inferior. Aquí tienes el código actualizado:

latex Copy code

**Tabla 3-1:** Tabla comparativa entre sensores

Número de Pin	Nombre del Pin	Descripción	Tipo
1	GND	Tierra	-
2	Vin	Voltaje de entrada a la placa, el cual puede ser mediante conexión USB, o una fuente externa	Puertos de alimentación
3	5V	Pin que produce un voltaje de 5V	Puertos de alimentación
4	3.3V	Pin que produce un voltaje de 3.3V	Puertos de alimentación
5	IOREF	Pin de referencia de voltaje con la que funciona el microcontrolador	Puertos de alimentación
6	RX	Pin 0 para recibir datos	Pines seriales
7	TX	Pin 1 para transmitir datos	Pines seriales
8	PWM	Pines 3, 5, 6, 9, 10 y 11, salida de 8 bits	-
9	SPI	Interfaz periférica en serie, pin 10, 11, 12, y 13	-
10	TWI	Pin A4 (SDA), y pin A5 (SCL)	-

**Realizado por:** Guillermo P., 2024

#### 3.4.1.2 Sensor cardíaco AD8232

El sensor AD8232 de la marca sparkfun, es uno de los sensores principales para el funcionamiento del prototipo, cuenta con la señal de salida analógico IC, siendo fácil la lectura por un microcontrolador, en un intervalo de PR y QT del latido del corazón, por lo que se requiere definir características adicionales, y la descripción de los pines, mostrados en la Tabla 4-1, que ayuda para identificar el comportamiento del sensor.

**Tabla 3-2:** Descripción de Pines del sensor AD8232

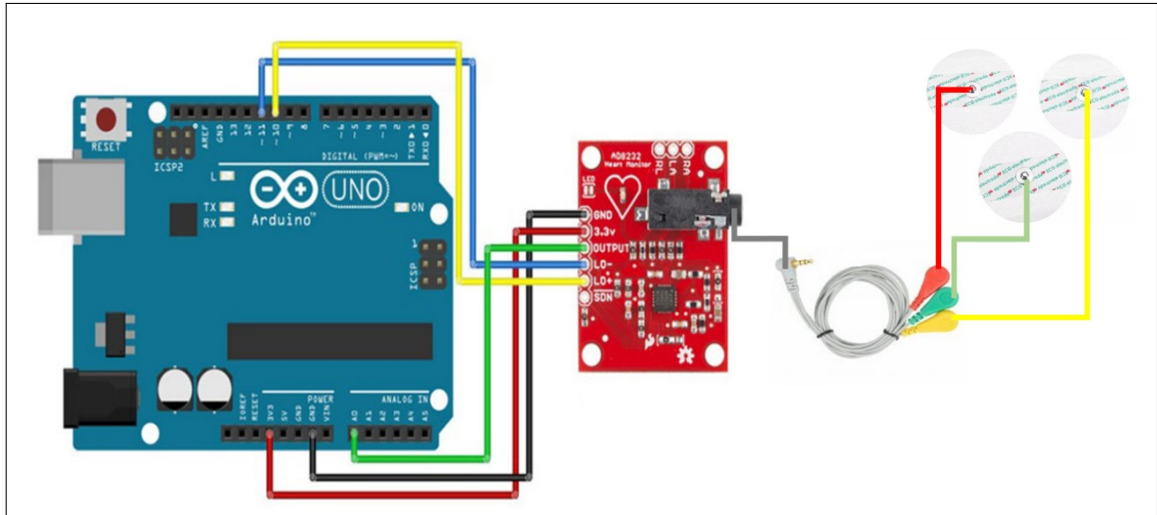
Número de Pin	Nombre del Pin	Descripción	Tipo
1	GND	Tierra	Puertos de poder
2	3.3V	Fuente de alimentación	-
3	RA (Right arm)	Entrada positiva del amplificador, el cual recibe la señal del electrodo conectado al brazo izquierdo	Puertos de los electrodos biométricos
4	LA (Left arm)	Entrada negativa del amplificador, el cual recibe la señal del electrodo conectado al brazo derecho	Puertos de los electrodos biométricos
5	RL (Right leg)	Señal del electrodo conectado a la pierna derecha	Puertos de los electrodos biométricos
6	3.5 mm female jack	Conexión del electrodo biométrico	-
7	Pin output	Puerto de salida de la señal amplificada y filtrada	Puerto de salida (amplificador)
8	LO-	Utilizado como referencia de salida del sensor	Puerto de comparación
9	LO+	Parte de la salida diferencial del sensor	-
10	~SND	Modo de apagado, utilizado para controlar la potencia de entrada	Shoutdown
11	LED	Indica el funcionamiento del sensor	-

Realizado por: Pinguil G., 2024

- La relación de rechazo en modo común es de 80 dB a 60 Hz
- Descarga electrostática para el cuerpo humano es de 8kV
- Detección de la conexión o no conexión de los cables
- Cuenta con un modo de ahorro de energía
- Temperatura nominal de 0 a 70 °C

- Temperatura de trabajo es de -40 a 85 °C

En la Ilustración 3-6 se describe las conexiones de los pines utilizados con el arduino uno, y los electrodos para la configuración del mismo.



**Ilustración 3-6:** Conexiones del sensor AD8232

Realizado por: Guillermo P, 2024

La configuración del sensor se realiza en bloques:

1. El primer bloque es la declaración de las variables y puertos a utilizar, como se muestra en la Ilustración 3-6.

```

1  const int analogInputPin = A0;    // Pin analógico de entrada
2  const int filteredOutputPin = 9;  // Pin de salida del filtro
3
4  const int numSamples = 30;        // Número de muestras para promediar
5  int samples[numSamples];          // Array para almacenar las muestras
6  int sampleIndex = 0;              // Índice de la muestra actual
7  int filteredValue = 0;            // Valor filtrado inicial
8  const float alpha = 0.7;          // Factor de suavizado (0 < alpha < 1)
9  const int hysteresis = 10;        // Valor de histeresis

```

**Ilustración 3-7:** Declaración de variables

Realizado por: Guillermo P, 2024

2. El segundo bloque es el void set up, que nos ayuda a la configuración de los pines de entrada y salida, así como las frecuencias que se va a utilizar con la finalidad de establecer una compatibilidad, como se muestra en Ilustración 3-7.

```

11 void setup() {
12   Serial.begin(9600);
13   pinMode(10, INPUT);           //PIN para la detección IO+
14   pinMode(11, INPUT);           //PIN para la detección IO-
15   pinMode(filteredOutputPin, OUTPUT); // Configurar el pin de salida del filtro
16 }

```

### Ilustración 3-8: Bloque void setup

Realizado por: Guillermo P., 2024

- El tercer bloque es el void loop, en la cual se realiza un filtrado adicional, para la recolección de la información más exacta sin la introducción de mucho ruido, ya que el sensor utilizado cuenta con la característica de introducir mucho ruido a la señal recibida, como se muestra en la Ilustración 3-8.

```

18 void loop() {
19   static int counter = 0;
20
21   if ((digitalRead(11) == 1) || (digitalRead(10) == 1)) {
22     if (counter % 1 == 0) {
23       Serial.println("%");
24     }
25   } else {
26     if (counter % 1 == 0) {
27       int sensorValue = analogRead(analogInputPin); // Leer el valor analógico de entrada
28       if (abs(sensorValue - filteredValue) > hysteresis) { // Aplicar histeresis
29         filteredValue = sensorValue; // Solo actualizar si la diferencia es mayor que el valor de histeresis
30       }
31
32       samples[sampleIndex] = filteredValue; // Almacenar la muestra en el array
33       sampleIndex = (sampleIndex + 1) % numSamples; // Actualizar el índice de la muestra
34       filteredValue = alpha * filteredValue + (1 - alpha) * sensorValue; // Calcular el promedio ponderado con un filtro de paso bajo
35       Serial.println(filteredValue); // Enviar el valor filtrado por el monitor serie
36     }
37   }
38
39   counter++;
40   delay(5);
41 }

```

### Ilustración 3-9: Bloque void loop

Realizado por: Guillermo P., 2024

#### 3.4.1.3 Módulo Bluetooth HC-06

El módulo Bluetooth HC-06 del fabricante OSSEPP Electronics, es el módulo utilizado para la transmisión de la información a la aplicación móvil, este módulo cuenta con una serie de características favorables:

- Cuenta con una antena y un transceptor inalámbrico digital de 2,4 GHz
- Circuito periférico simple, con saltos de frecuencia adaptativos
- Cuenta con un USB 1.1 de máxima velocidad, y es compatible con el protocolo USB 2.0

- Sensibilidad de  $-80$  dBm y potencia de salida de  $-4$  dBm a 6 dBm
- Potencia de emisión de 3 dBm
- Cobertura de 10 metros (Clase 2)
- Protocolo RFCOMM, o también conocido como emulador de puerto serial.

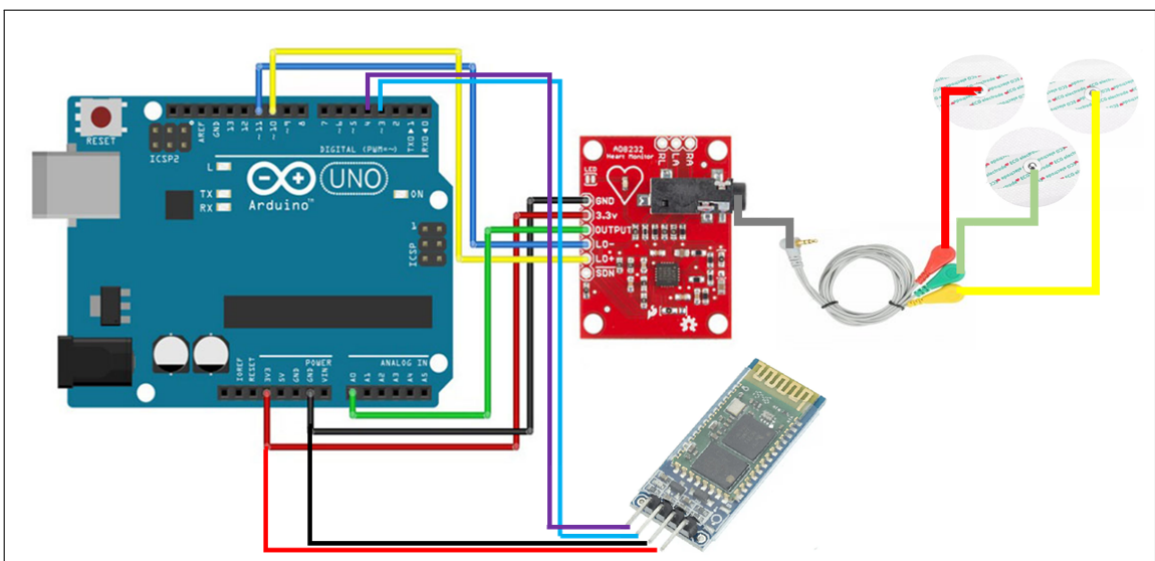
Cuenta con cuatro pines y componentes adicionales que se describe en la Tabla 3-4

**Tabla 3-3:** Descripción de Pines y componentes del módulo Bluetooth HC-06

Número de Pin	Nombre del Pin	Descripción	Tipo
1	GND	Tierra	Puertos de poder
2	VCC	Fuente de alimentación	
3	TX	Pin de transmisión de datos	Pines de comunicación
4	RX	Pin de recepción de datos	
5	KEY	Pone a nivel alto para entrar en configuración del módulo	Pines opcionales
6	STATE	Conexión de un led de salida para visualizar la comunicación exitosa	Pines opcionales

Número del componente	Nombre	Descripción	Tipo
1	LED	Indica la conexión de Bluetooth	Conectividad
2	Antena	Incorporada en el PCB	-

*Realizado por: Guillermo P., 2024*



**Ilustración 3-10:** Conectividad del módulo Bluetooth

Realizado por: Guillermo P., 2024

Para la configuración del módulo Bluetooth se utiliza el código para la configuración del sensor AD8232 como base, siendo la señal filtrada por el sensor la información que se va a enviar mediante el Bluetooth a la aplicación móvil, para ello se añade líneas de comando las cuales están señaladas en la Ilustración 3-10, la primera línea de comando es utilizada para la declaración de la librería, el segundo comando es para la definición de los puertos RX y Tx para la conexión del módulo a la tarjeta de desarrollo, se inicializa el puerto serial a 9600 Hz en el comando tres, y el envío de la información se realiza mediante el comando cuatro añadido.

```

1 #include <SoftwareSerial.h>
2
3
4 const int analogInputPin = A0; // Pin analógico de entrada
5 const int filteredOutputPin = 9; // Pin de salida del filtro
6 const int numSamples = 30; // Numero de muestras para promediar
7 int samples[numSamples]; // Array para almacenar las muestras
8 int sampleIndex = 0; // Índice de la muestra actual
9 int filteredValue = 0; // Valor filtrado inicial
10 const float alpha = 0.7; // Factor de suavizado (0 < alpha < 1)
11 const int hysteresis = 1; // Valor de histéresis
12 SoftwareSerial bluetoothSerial(3, 4); // RX, TX para Bluetooth
13
14 void setup() {
15   Serial.begin(9600);
16   bluetoothSerial.begin(9600);
17   pinMode(11, INPUT);
18   pinMode(filteredOutputPin, OUTPUT); // Configurar el pin de salida del filtro
19 }
20
21 void loop() {
22   static int counter = 0;
23   if ((digitalRead(11) == 1) || (digitalRead(10) == 1)) {
24     if (counter % 1 == 0) {
25       Serial.println("AV");
26     } else {
27       if (counter % 1 == 0) {
28         int sensorValue = analogRead(analogInputPin); // Leer el valor analógico de entrada
29         if (abs(sensorValue - filteredValue) > hysteresis) { // Aplicar histéresis
30           filteredValue = sensorValue; // Solo actualizar si la diferencia es mayor que el valor de histéresis
31         }
32         samples[sampleIndex] = filteredValue; // Almacenar la muestra en el array
33         sampleIndex = (sampleIndex + 1) % numSamples; // Actualizar el índice de la muestra
34         filteredValue = alpha * filteredValue + (1 - alpha) * sensorValue; // Calcular el promedio ponderado con un filtro de paso bajo
35         //Serial.println(filteredValue);
36         bluetoothSerial.println(filteredValue);
37       }
38       counter++;
39       delay(5);
40     }
41   }
42 }

```

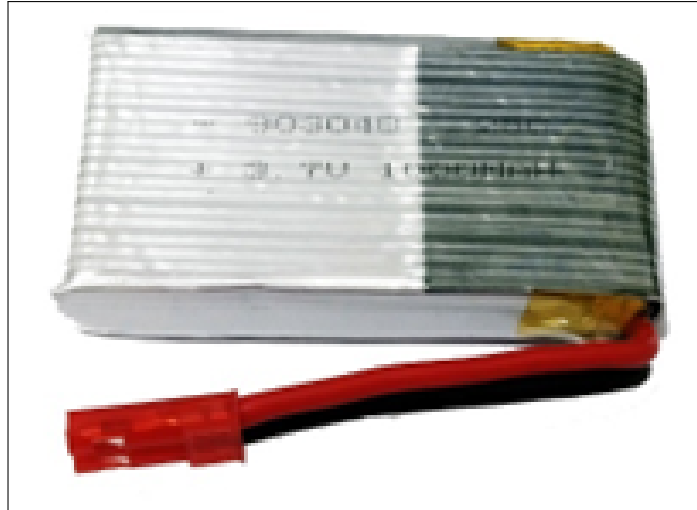
**Ilustración 3-11: Código del módulo Bluetooth**

Realizado por: Guillermo P., 2024

### 3.4.1.4 Fuente de alimentación

Para poder dar energía al electrocardiógrafo se utiliza una batería recargable lipo de 3.7 V. cual cuenta con un puerto micro USB para recarga, la batería es pequeña y liviana para la que el prototipo sea cómodo de llevar para el paciente.





**Ilustración 3-12:** Batería lipo.

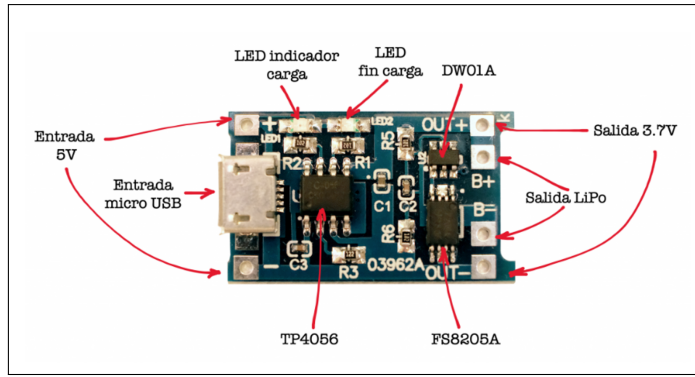
Realizado por: Guillermo P., 2024

### Características

- Batería de Lipo 25C
- 3.7V
- 1000mAh
- 903048 Litio
- Largo: 48mm
- Ancho: 30mm
- Alto: 9mm

La batería lipo es utilizada para dar energía al prototipo, ya que la batería solo contiene dos polos, el polo positivo que da 3.7V y el polo negativo para la conexión a tierra (GND), la conexión directa a la tarjeta de desarrollo no es posible por lo que se requiere de componentes adicionales para su conexión, el módulo TP4056, y el módulo MT3608.

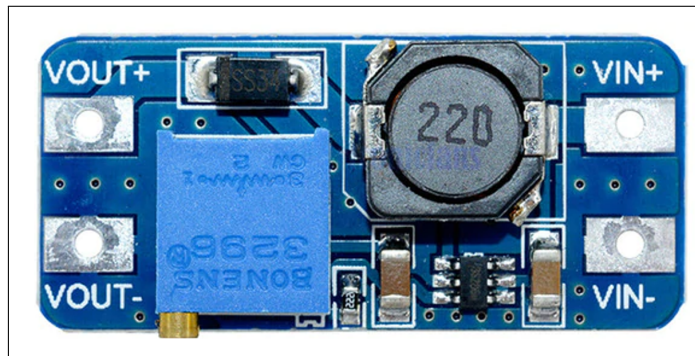
El módulo TP4056, se trata de un controlador de voltaje, que ayuda como cargador programable lineal de corriente y tensión constante para baterías lipo de celda única, cuenta con dos leds indicadores del estado de la carga y dos chips adicionales (DW01A-FS8205A) encargados de la protección de la batería, dichos pines son descritos en la Ilustración 3-13.



**Ilustración 3-13:** Pines del módulo TP4056

Fuente: (BIKEPICSELS, 2019)

El convertidor de voltaje DC-DC MT3608 de la Ilustración 3-13, cumple con la función de entregar voltajes de salida superior al voltaje de entrada, soporta corrientes de salida de hasta 2A, voltajes de entrada entre 2V a 24V y voltajes de salida de 2 a 28V.



**Ilustración 3-14:** Convertidor MT3608

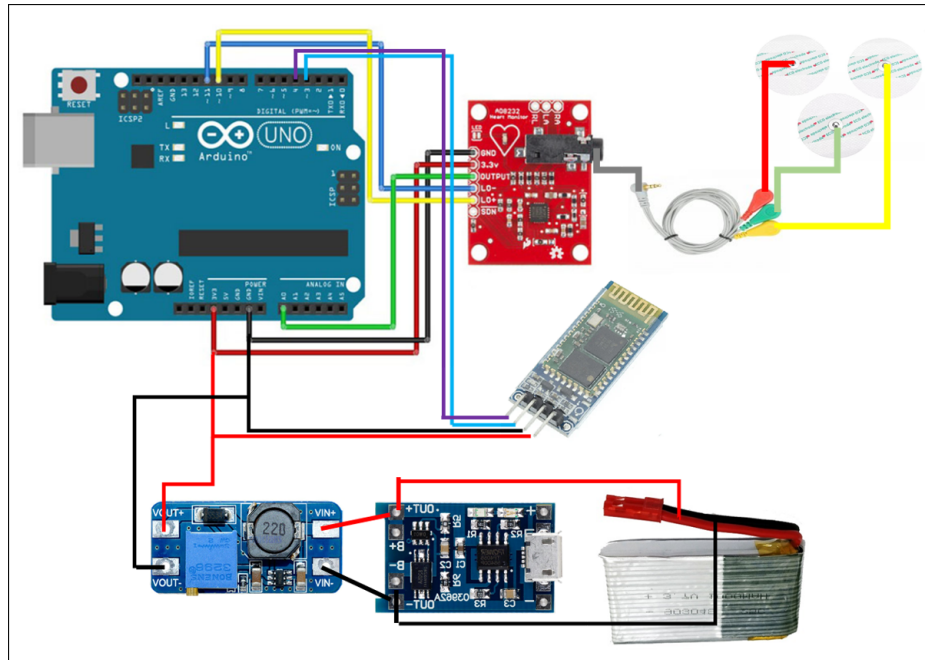
Fuente: (ROBOTS-ARGENTINA, 2023)

Los pines del convertidor MT3608 están descritos en la Tabla 3-4.

**Tabla 3-4:** Descripción de Pines y componentes del módulo Bluetooth HC-06

Número de Pin	Nombre del Pin	Descripción	Tipo
1	Vout+	Pin de salida polo positivo	Puertos de entrada
2	Vout-	Pin de salida polo negativo	
3	Vin+	Pin de entrada polo positivo	Puertos de salida
4	Vin-	Pin de entrada polo negativo	

Realizado por: Guillermo P., 2024

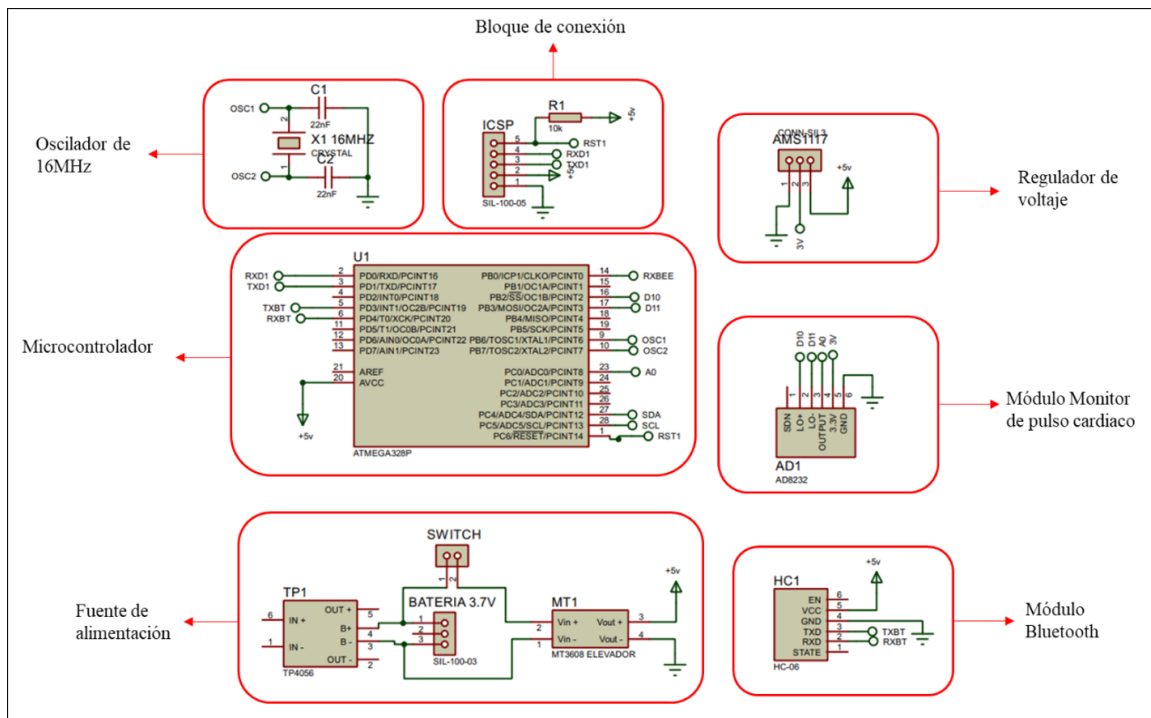


**Ilustración 3-15:** Conexiones del prototipo con la Batería lipo.

Fuente: (ROBOTS-ARGENTINA, 2023)

Se procede al diseño del circuito electrónico, con todos los componentes necesarios para el funcionamiento del prototipo, en la Ilustración 3-16 se visualiza el esquema electrónico, como resultado final de todos los componentes que incluye el prototipo, entre los principales componentes se tiene:

- Atmega328P: es un microcontrolador.
- Switch que permite el funcionamiento del prototipo, ya que está conectado a una batería de 3.7V.
- Módulo Bluetooth HC-06.
- Módulo monitor de pulso cardíaco AD8232.
- Oscilador de cristal de 16MHz para controlar el tiempo, que se caracteriza por su estabilidad de frecuencia y pureza de fase.

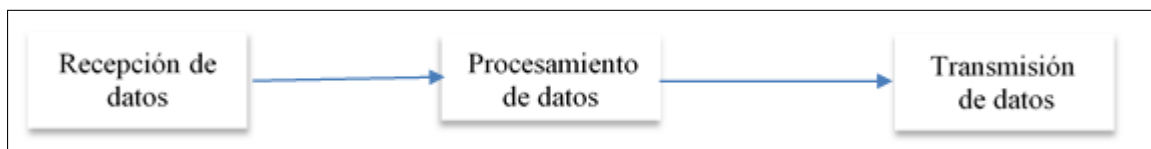


**Ilustración 3-16:** Esquema electrónico

Realizado por: Guillermo P., 2024

### 3.4.2 Nodo receptor

El diagrama de bloque del nodo receptor está compuesto por 3 bloques Ilustración 3-17, en un inicio se muestra se muestra un bloque de recepción de datos en el dispositivo móvil, estos datos serán procesados y se muestra una gráfica de la frecuencia cardiaca en el dispositivo móvil. Finalmente, se muestra el bloque encargado de enviar todos estos datos a la nube, para que pueda ser evaluada por un profesional de la salud y revise el comportamiento del órgano cardiaco del paciente.



**Ilustración 3-17:** Diagrama de bloques del sistema, nodo transmisor receptor

Realizado por: Guillermo P., 2024

En la Ilustración 3-18 se establece un subproceso dentro del nodo receptor, el cual consiste en el registro del número de cedula del paciente, el cual se utilizó para identificar a quien le corresponden los datos almacenados en la base de datos de la nube. Esto se realiza desde la aplicación móvil, para posteriormente ser enviada a la nube y se pueda visualizar

desde una página web.



**Ilustración 3-18:** Diagrama de bloques del sistema, módulo de registro

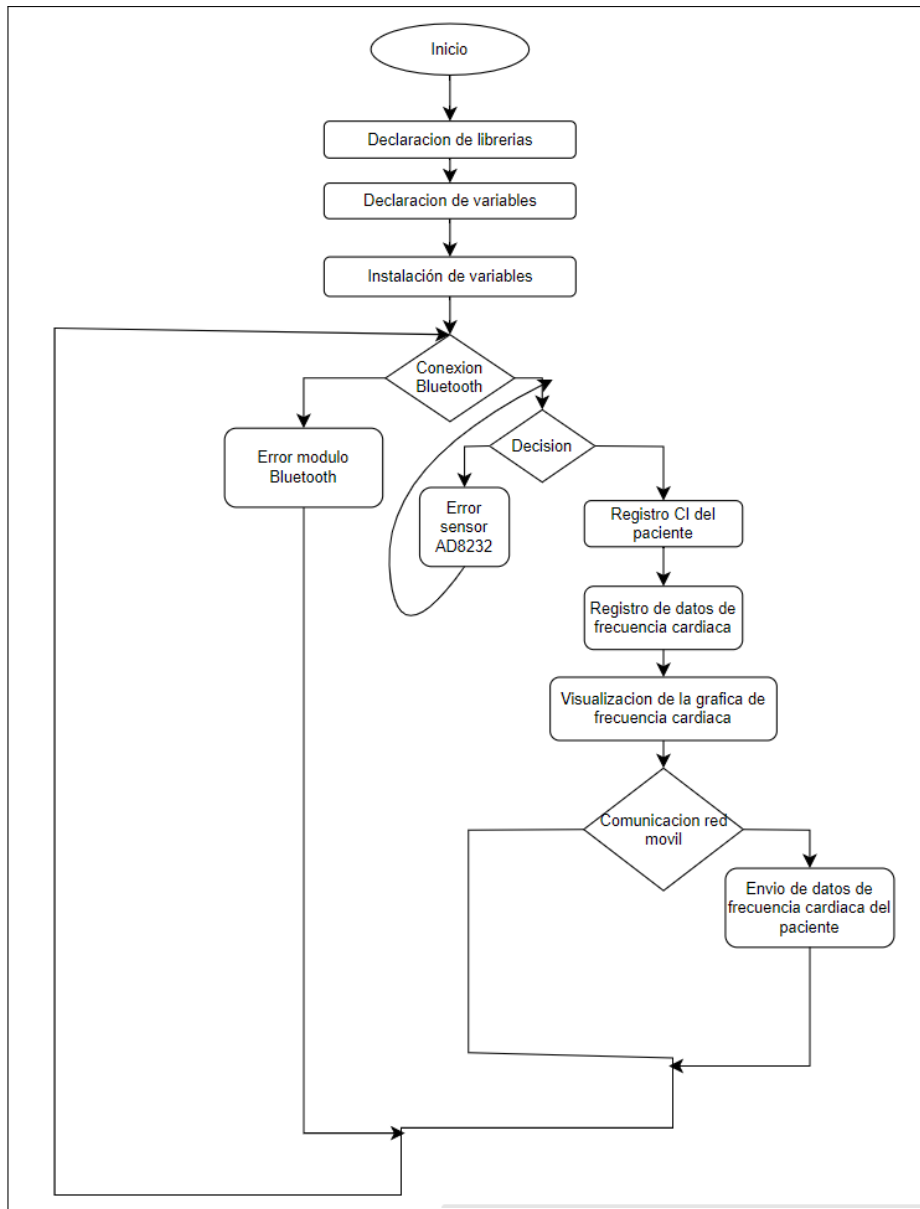
Realizado por: Guillermo P., 2024

#### 3.4.2.1 Aplicación móvil

Uno de los componentes software del proyecto es la aplicación móvil, que cuenta con varias características que ayudan a la comunicación, así como el manejo fácil del mismo, se estableció que debe de tener diferentes opciones, las cuales son descritas a continuación:

- Selección del módulo Bluetooth
- Tener una comunicación constante con el módulo Bluetooth para la recepción de datos en tiempo real.
- Tener una comunicación constante ya sea mediante Wi-Fi o LTE, para el envío de la información a la página web en tiempo real.
- Ingreso de la información personal del paciente como nombre y cédula.
- Visualización de los datos obtenidos por el módulo Bluetooth, tanto de manera numérica como gráfica.

El funcionamiento y componentes de la aplicación móvil se define en el diagrama de flujo de la Ilustración 3-19



**Ilustración 3-19:** Diagrama de flujo del sistema, módulo de registro.

Realizado por: Guillermo P., 2024

Una vez establecido el comportamiento, los parámetros y requisitos necesarios para el funcionamiento de la aplicación móvil, se procede a realizar el código mediante la plataforma Android Studio con el lenguaje de programación kotlin, siendo este un lenguaje moderno que ayuda a aumentar la productividad, cubre las necesidades de los desarrolladores y proporciona una seguridad del código. En el Anexo B: Código de aplicación móvil, se proporciona el código completo utilizado para la elaboración de la aplicación móvil para el presente proyecto.

Existe una serie de comandos para la implementación, y son necesarios su entendimiento,

los cuales se describen en la Tabla 3-5.

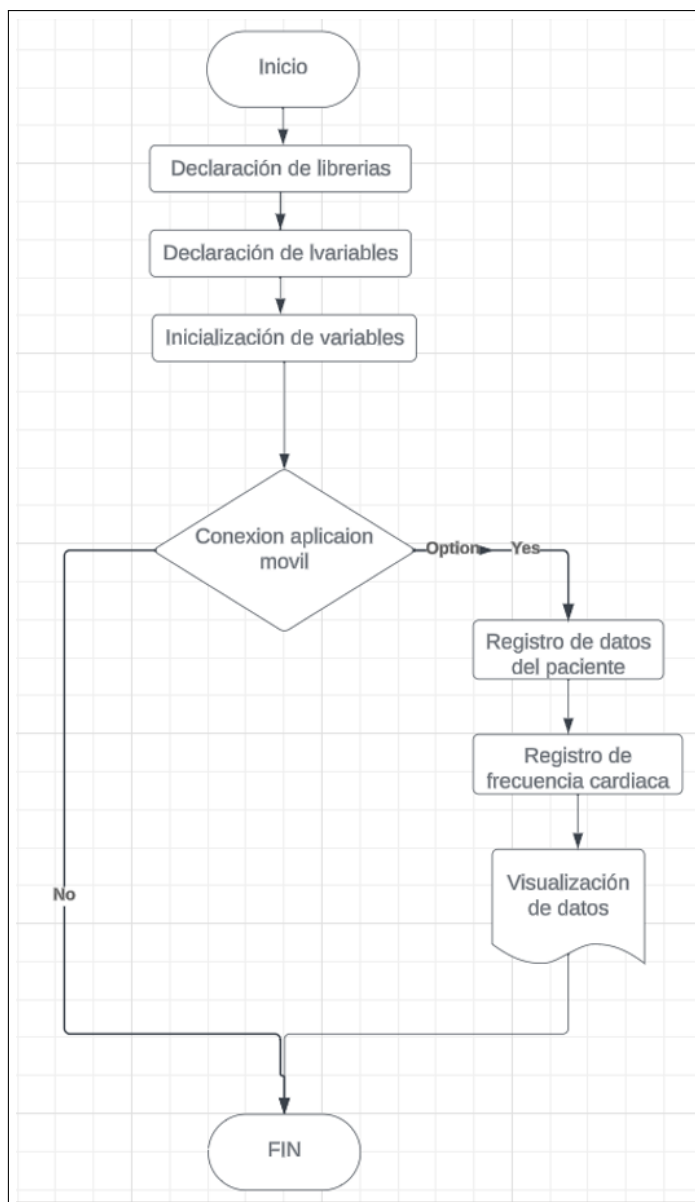
**Tabla 3-5:** Comandos de la aplicación móvil

<b>Comando</b>	<b>Descripción</b>
Package	Contiene la información y ubicación de la versión sobre la implementación
Import	Define los paquetes con los que se va a trabajar, los cuales contienen clases y funciones
val, var	Define variables con su nombre, pueden ser numéricas, alfanuméricas o booleanas. 'val' no se pueden modificar en el tiempo, y 'var' si se modifica en el tiempo
Class	Define un tipo de dato
Fun	Define una función
companion object	Declarar propiedades sujetas a un tipo y no a un objeto en concreto
Chart	Instrucción para graficar
textView()	Presentación de un texto al usuario
onCreate()	Inicialización única para una actividad
If() else()	Instrucción de condición para devolver un resultado específico
Toast.makeText()	Widget de interfaz de usuario que aparece luego de una modificación
Set	Define un método de descriptor de acceso a una propiedad
Result	Forma de manejo de un resultado de una operación
ArrayAdapter	Herramienta para vincular fácilmente una lista de datos a una vista de datos
findViewById	Vista de un ID en el layout de una actividad
While	Bucle para devolución de un resultado
For	Bucle para devolución de un resultado

Realizado por: Guillermo P., 2024

#### 3.4.2.2 *Página web*

En el receptor uno de los componentes más importantes es la página web, en la Ilustración ?? se visualiza el diagrama de flujo del funcionamiento de la página web, esta página web refleja todos los datos guardados y en tiempo real que se están transmitiendo y se han transmitido desde la aplicación móvil.



**Ilustración 3-20:** Diagrama de flujo de la página web.

Realizado por: Guillermo P., 2024

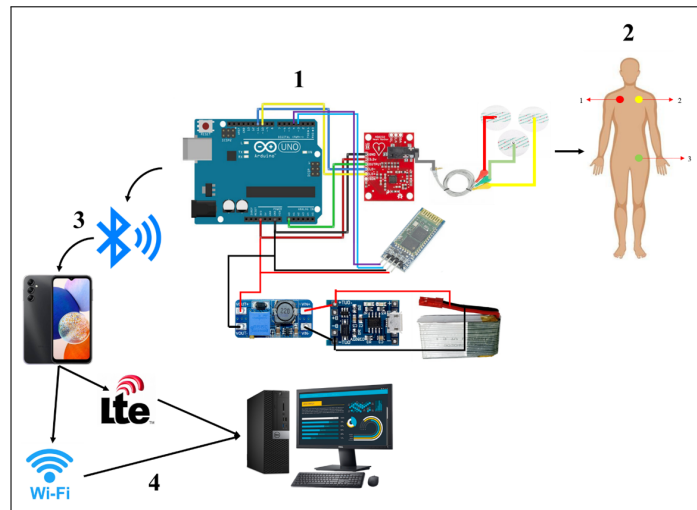
Para la implementación de la página web, se utiliza la plataforma Visual Studio Code, con el lenguaje de programación kotlin, permitiendo la compatibilidad entre la aplicación móvil y la página web, se utiliza una serie de comandos para la implementación, y están descritos en el Anexo C: Código página web.

### 3.5 Toma de resultados

El esquema general de los componentes y conexiones necesarias para el funcionamiento del prototipo se define en la Ilustración 3-21, la cual está conformado por bloques que



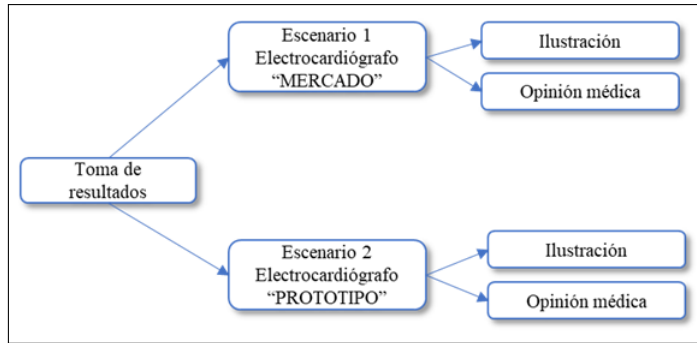
están relacionados entre sí, el bloque 1 se establece la recolección de la información del paciente, siendo este el bloque 2, conectado a través de electrodos en posiciones específicas, los datos recolectados en conjunto de estos dos bloques son enviados mediante Bluetooth al bloque 3 es decir la aplicación móvil, en la que se visualiza los resultados y se registra los datos del paciente, para ser enviados al bloque 4 es decir la página web, el envío de la información cumple con ciertas especificaciones, definidas en el bloque 3, la cuales pueden ser transmitidas mediante Wi-Fi o mediante LTE.



**Ilustración 3-21:** Esquema general del prototipo.

Realizado por: Guillermo P., 2024

Una vez se haya establecido el funcionamiento del prototipo, se procede a la validación del mismo mediante la recolección de datos, estos datos son considerados para dos diferentes escenarios, el primer escenario se considera la toma de resultados con un electrocardiógrafo de mercado, y el segundo escenario se considera los resultados del electrocardiógrafo prototipo de dicho trabajo de titulación, para la validación se realiza mediante una comparativa entre los 2 dispositivos, proceso que se describe en la Ilustración 3-22, para este proceso se considera la recolección y comparación de los resultados de dos pacientes diferentes.



**Ilustración 3-22:** Diagrama de bloque de la toma de resultados.

Realizado por: Guillermo P., 2024

## CAPÍTULO IV

### 4. RESULTADOS

En el presente capítulo se define las diferentes etapas para la construcción del prototipo, comenzando con el diseño en el que se detalla cada uno de los componentes, y formas de visualización, entre estos se encuentra la aplicación móvil y la página web, considerando la forma de comunicación con el prototipo.

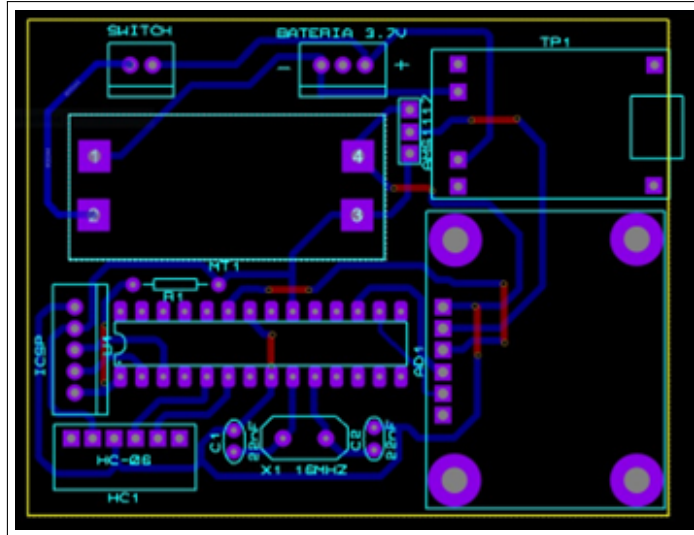
#### 4.1 Etapa 1: Diseño

En la etapa de diseño se establece los componentes y recursos necesarios para establecer el funcionamiento del prototipo, dentro de esta etapa se considera el diseño de la placa, en donde se encuentran los diferentes componentes para el funcionamiento del prototipo, y el módulo de comunicación, que es utilizado para el envío de la información a la aplicación móvil, y a su vez a la página web (nube).

##### 4.1.1 *Diseño de la placa y conexiones*

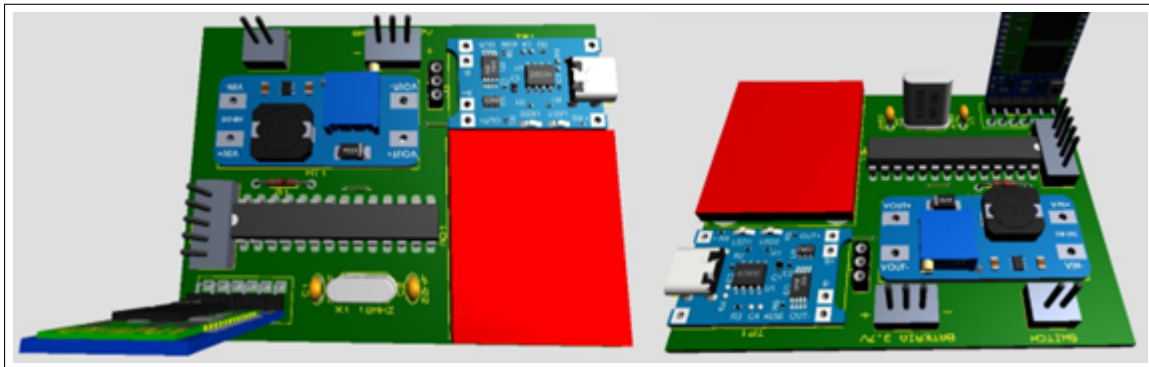
Para el diseño de la placa se utilizó un software CAD de automatización de diseño electrónico, el cual cuenta con una serie de herramientas, entre ellas se encuentra el diseño de PCB, que permite escalar el tamaño de las placas que pueden ser construidas, simulando en conjunto todos los componentes, así como la depuración de errores.

Una vez que se obtiene el esquema electrónico, que funciona correctamente, se procede al diseño del PCB, en donde cada uno de los bloques del esquema electrónico es ubicado estratégicamente de tal forma se tenga una idea de la conectividad de los componentes, así como se describe en la Ilustración 4-1.



**Ilustración 4-1: Diseño PCB**  
Realizado por: Guillermo P., 2024

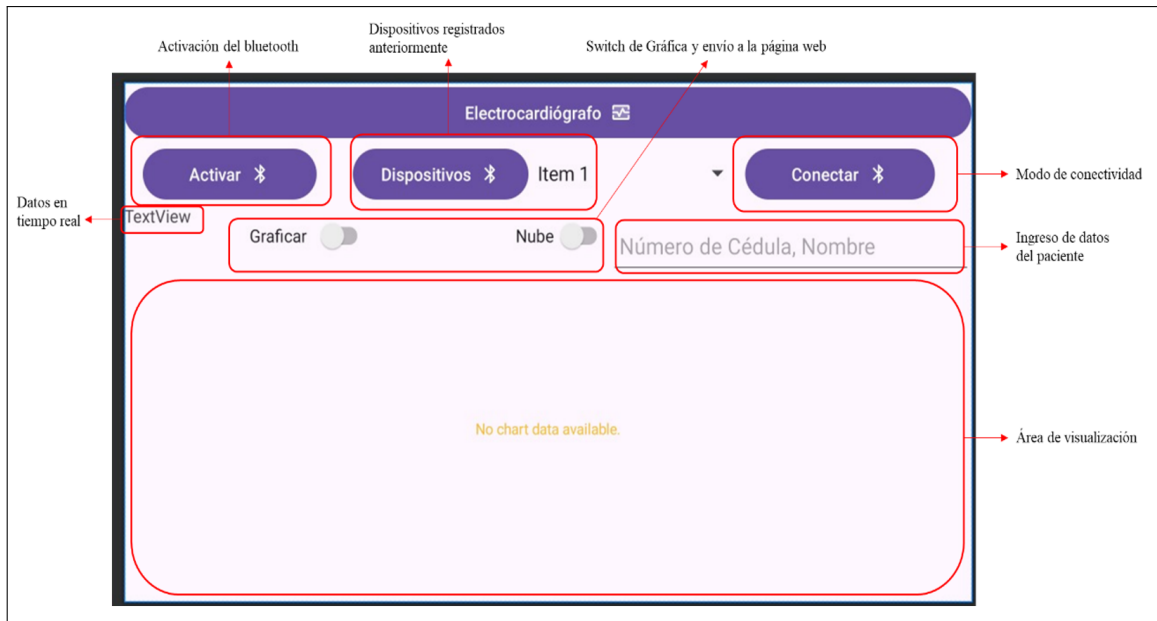
Una de las herramientas que permite el software CAD de diseño de circuitos es la visualización en 3D, donde se tiene una idea del diseño final de la placa, considerando todos los componentes, la visualización 3D del prototipo se describe en la Ilustración 4-2.



**Ilustración 4-2: Visualización 3D**  
Realizado por: Guillermo P., 2024

#### 4.1.2 *Diseño de la aplicación móvil*

Para el diseño de la aplicación móvil se consideró el entorno de desarrollo Android Studio, con un lenguaje comunicación kotlin, el cual permite un fácil manejo, y como objetivo del trabajo de titulación permite la conectividad mediante Bluetooth o LTE. En la Ilustración 4-3 se visualiza la configuración de la aplicación móvil.



### Ilustración 4-3: Componentes de la aplicación móvil

Realizado por: Guillermo P., 2024

A continuación la descripción de los componentes de la aplicación móvil.

- **Activación de Bluetooth:** Permite la activación de Bluetooth, para lo cual la aplicación solicita permisos, como se visualiza en la Ilustración 4-4.

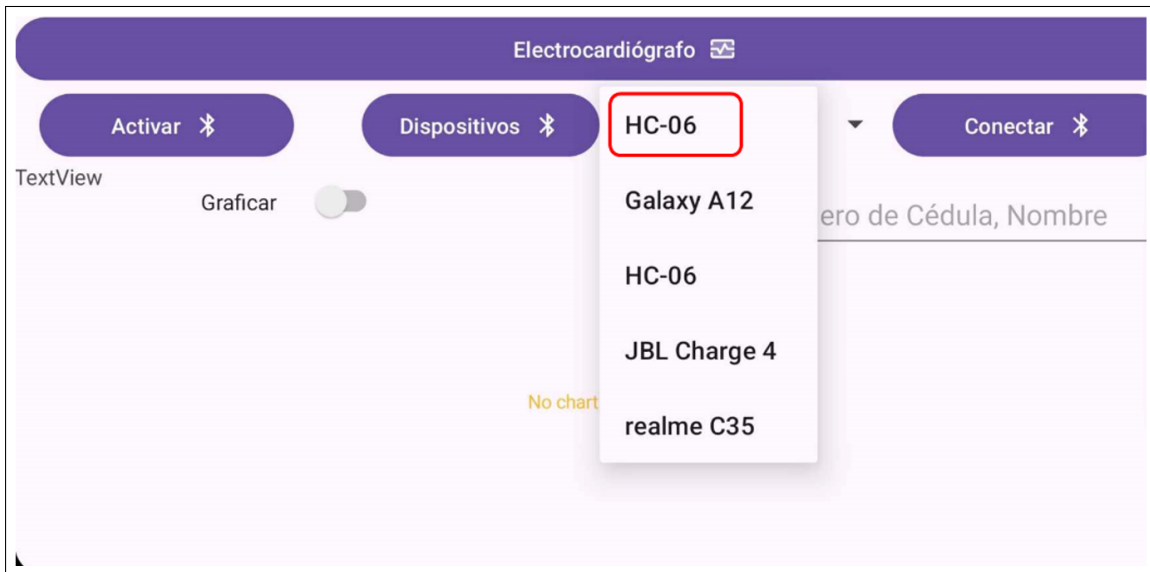


### Ilustración 4-4: Activación de Bluetooth

Realizado por: Guillermo P., 2024

- **Dispositivos registrados:** permite la conectividad, por Bluetooth a dispositivos vinculados con anterioridad, en la Ilustración 4-6 selecciona el módulo HC-06, el cual

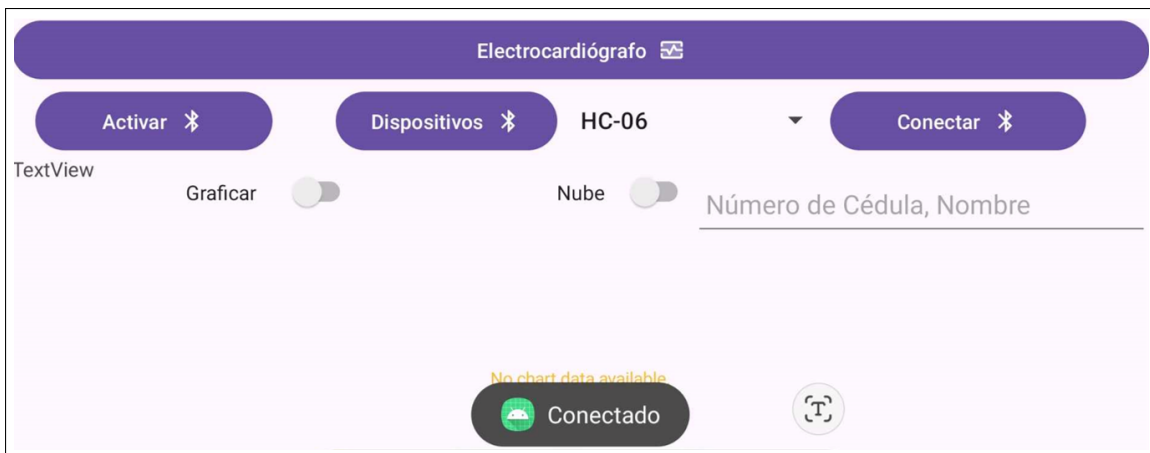
permite la conectividad con el prototipo.



#### Ilustración 4-5: Conectividad Módulo Bluetooth

Realizado por: Guillermo P., 2024

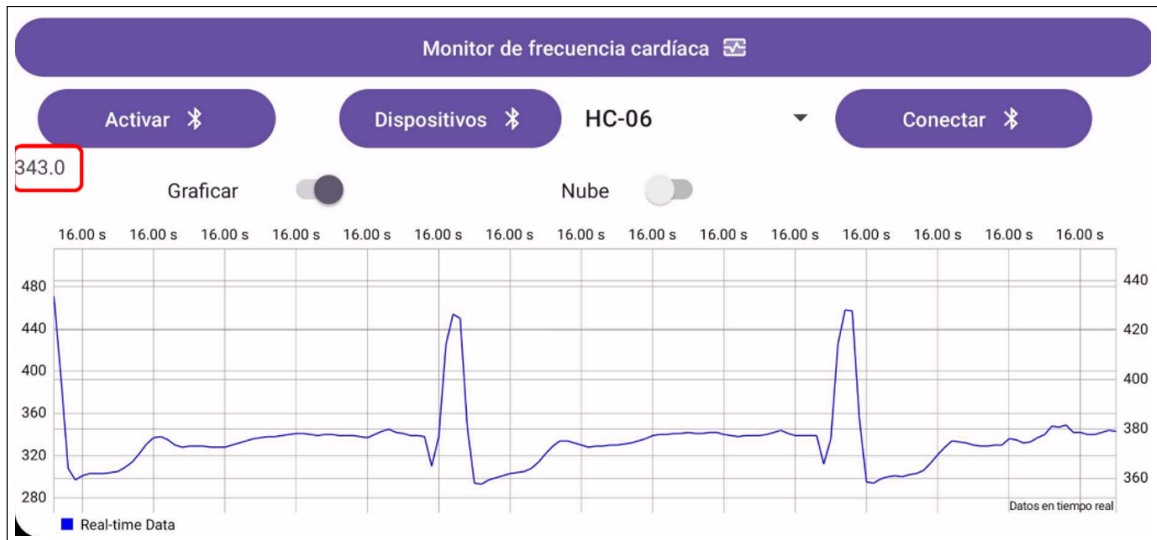
- Botón Conectar: una vez seleccionado el módulo Bluetooth HC-06, se presiona el botón Conectar, y para verificar la conectividad se espera el mensaje Conectado como se visualiza en la Ilustración 4-6.



#### Ilustración 4-6: Mensaje de conectividad

Realizado por: Guillermo P., 2024

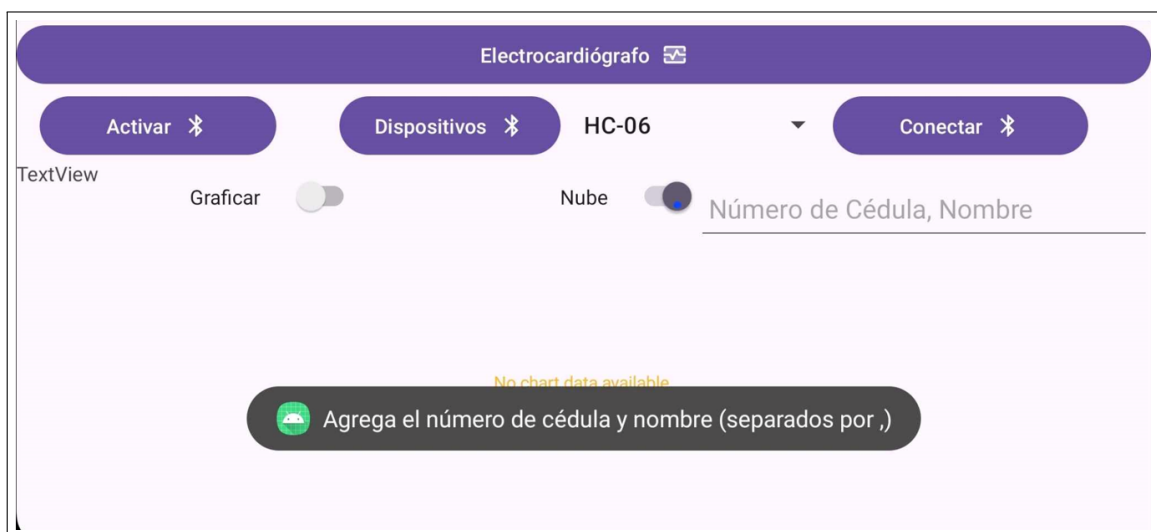
- TextView: se visualiza en tiempo real los datos que se obtienen del prototipo mediante Bluetooth.



**Ilustración 4-7:** Datos en tiempo real

Realizado por: Guillermo P., 2024

- Registro de datos: permite el registro de la información personal del paciente, como se describe en la Ilustración 4-8.



**Ilustración 4-8:** Datos en tiempo real

Realizado por: Guillermo P., 2024

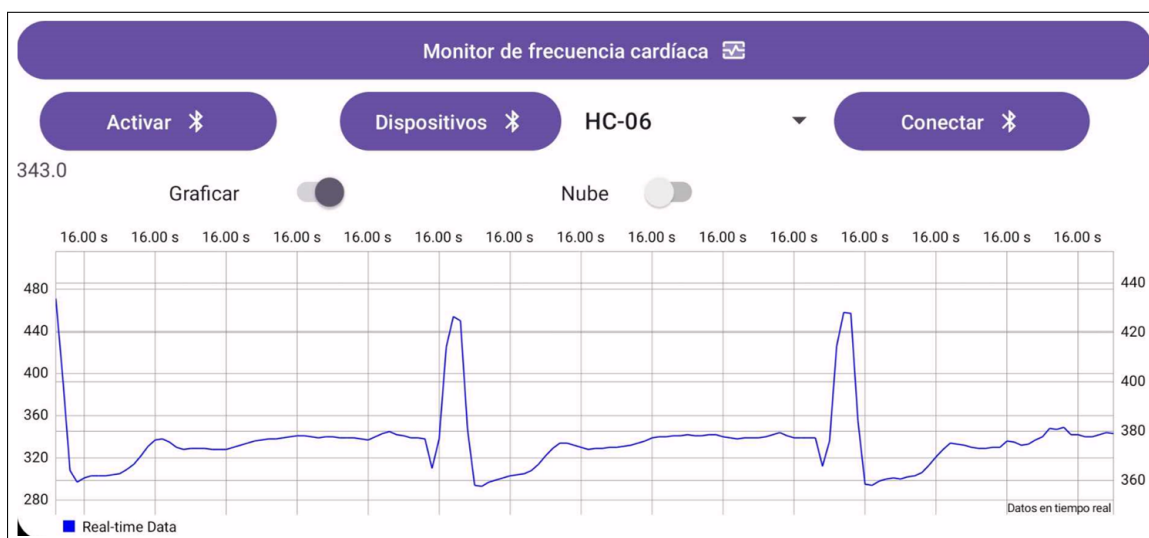
- Switches: cuenta con dos switch, el switch de graficar, el cual permite la visualización de los datos a lo largo del tiempo, y el switch de Nube, que permite el envío de la información a la Nube, y para verificar el envío de la información, se espera los mensajes de verificación, como se describe en la Ilustración 4-9.



**Ilustración 4-9:** Mensajes de verificación de envío a la nube

Realizado por: Guillermo P., 2024

- Área de visualización: Sección donde se grafica en tiempo real los resultados obtenidos, tomando en consideración que el switch Graficar debe estar encendido, como se describe en la Ilustración 4-10.



**Ilustración 4-10:** Área de visualización

Realizado por: Guillermo P., 2024

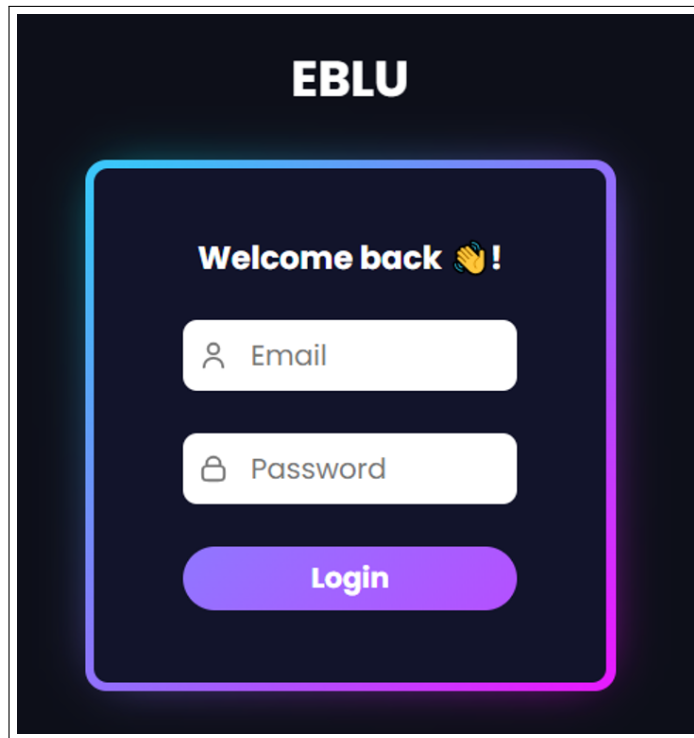
### 4.1.3 Diseño de la página web

Para el diseño de la página web se utilizó la aplicación visual studio code, con el lenguaje de programación JavaScript, compatible con cualquier sistema operativo, y para el diseño de la página web se considera el uso de seguridad para el ingreso a la información recopilada, como se visualiza en la Ilustración 4-11, se define a continuación el link de ingreso a la plataforma, así como las credenciales de prueba.



Usuario: alejandronov98@gmail.com

Contraseña: Guille98



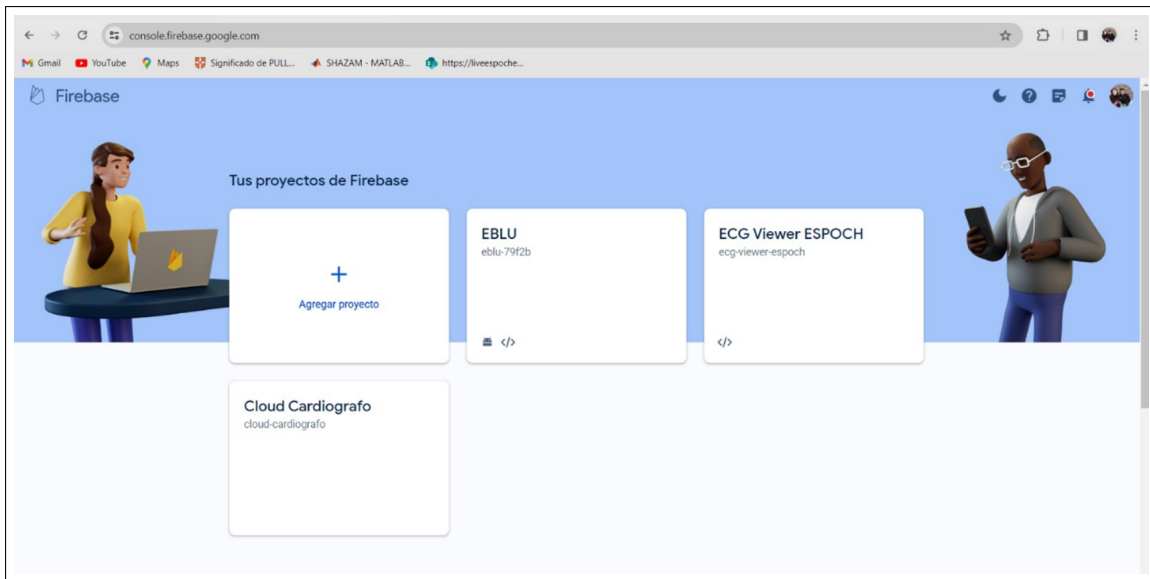
**Ilustración 4-11:** Seguridad página web

Realizado por: Guillermo P., 2024

Para el registro de credenciales nuevas para el ingreso a la página web, se realiza dentro de la plataforma digital Firebase, diseñada para facilidad en el desarrollo de aplicaciones web y móviles, además proporciona servicios integrados en relación a la seguridad, para el diseño del prototipo se define la autenticación de usuarios segura, utilizando correos electrónicos/contraseñas, Google, Facebook, entre otros.

Para el ingreso de nuevas credenciales, usados para la autorización de acceso a los datos sigue una serie de pasos descritos a continuación.

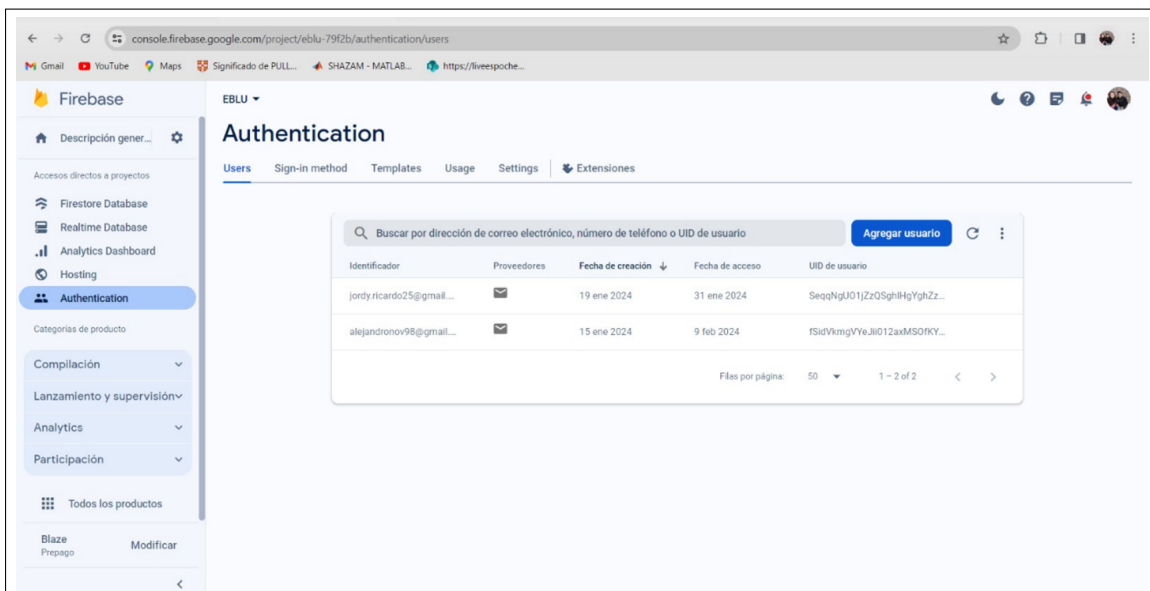
1. Se ingresa a la plataforma Firebase, en el que se encuentra el proyecto subido con anterioridad. En relación al trabajo de titulación, el proyecto se denomina EBLU, como se describe en la Ilustración 4-12.



**Ilustración 4-12:** Plataforma Firebase

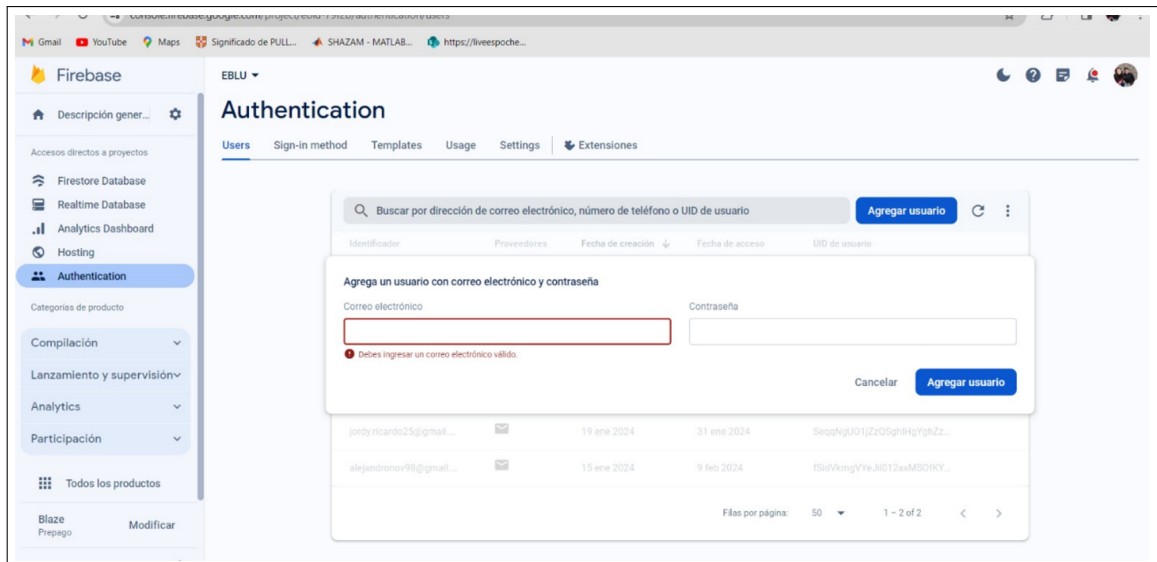
Realizado por: Guillermo P., 2024

2. En el menú lateral se selecciona Autenticación , en donde se configura el método de autenticación que se requiere utilizar, como correo electrónico/contraseña, Facebook, Google, entre otras opciones. Para el desarrollo del prototipo se selecciona correo electrónico/contraseña, en la Ilustración 4-13, se definen las credenciales registradas con anterioridad, y en la Ilustración 4-14 se define el proceso de registro de nuevas credenciales.



**Ilustración 4-13:** Credenciales cargadas en Firebase

Realizado por: Guillermo P., 2024

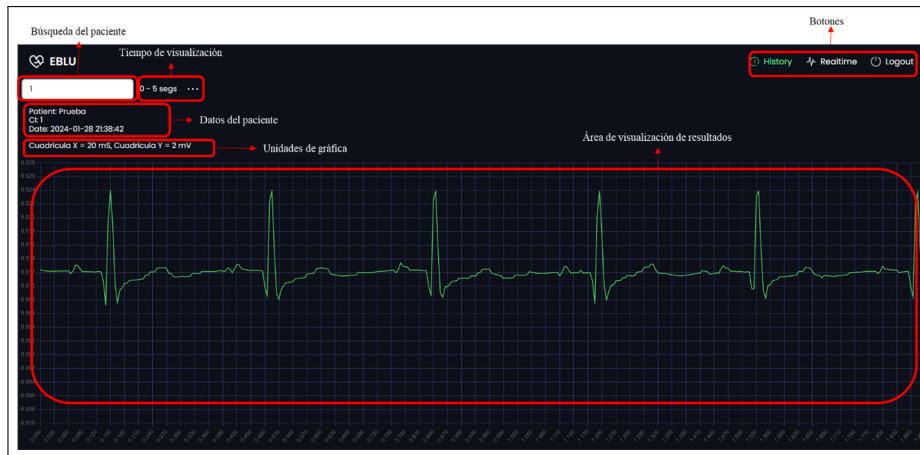


#### **Ilustración 4-14:** Proceso de nuevas credenciales en Firebase

**Realizado por:** Guillermo P., 2024

Una vez que se accede con las credenciales, en la Ilustración 4-15 se define los componentes que cuenta la página web, las cuales cuentan con:

- Barra de búsqueda: permite la visualización de los pacientes registrados previamente en la página web.
- Información: visualiza los datos personales del paciente seleccionado a través de la barra de búsqueda.
- Tiempo de visualización: tiempo que varían de 0-5 seg, 5-10 seg, 10-15 seg, 15-20 seg, 20-25 seg, y 25-30 seg, dependiendo del tiempo de muestra del paciente, dicho tiempo de visualización será considerado dependiendo de las necesidades del médico.
- Unidades de los resultados: unidades de los resultados obtenidos, en la coordenada X en mS y en la coordenada Y en mV.
- Área de visualización: gráfica de los datos obtenidos considerando una el tiempo de visualización.
- Botones de acceso: ingreso al historial, y establece la conectividad si existe una pérdida de conectividad, y salida del sistema.



**Ilustración 4-15:** Componentes página web

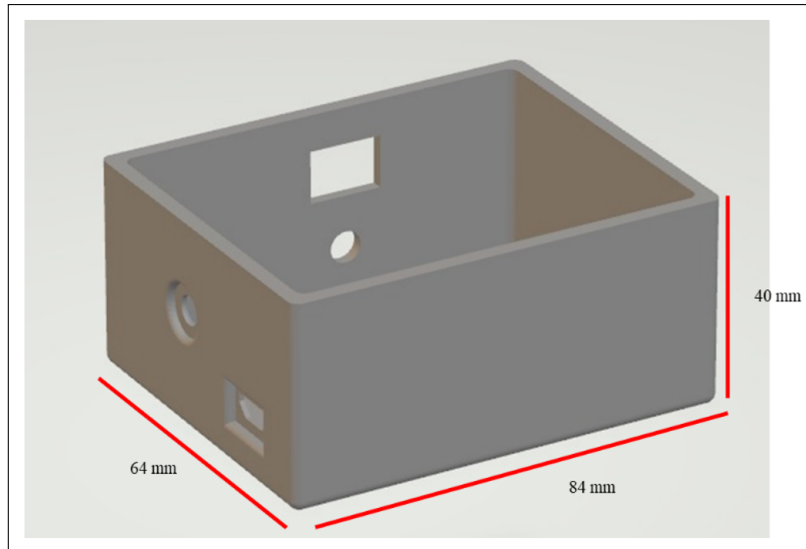
Realizado por: Guillermo P., 2024

## 4.2 Etapa 2: Diseño del protector plástico

La protección del prototipo es necesaria para extender el tiempo de vida útil del mismo, así como en la facilidad en el manejo del prototipo, se utiliza el software de diseño CAD el cual es un software de diseño, utilizado en diferentes industrias, para el diseño de productos, fabricación, arquitectura, y para el modelado en 3D .

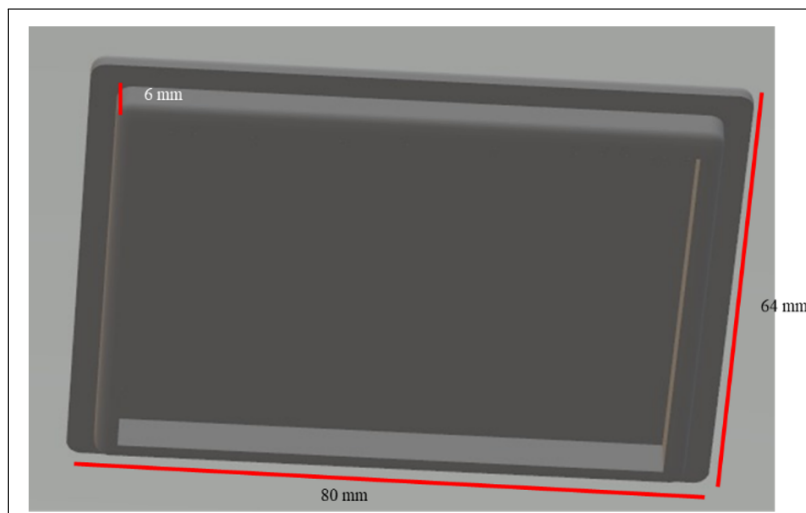
Para el prototipo se divide en dos partes:

- La primera parte es la carcasa que lleva la mayor parte del prototipo, con las aberturas tanto para las conexiones de los electrodos, el puerto USB y de carga de la batería, con las medidas descritas en la Ilustración 4-16.



**Ilustración 4-16:** Mediciones de la carcasa  
Realizado por: Guillermo P., 2024

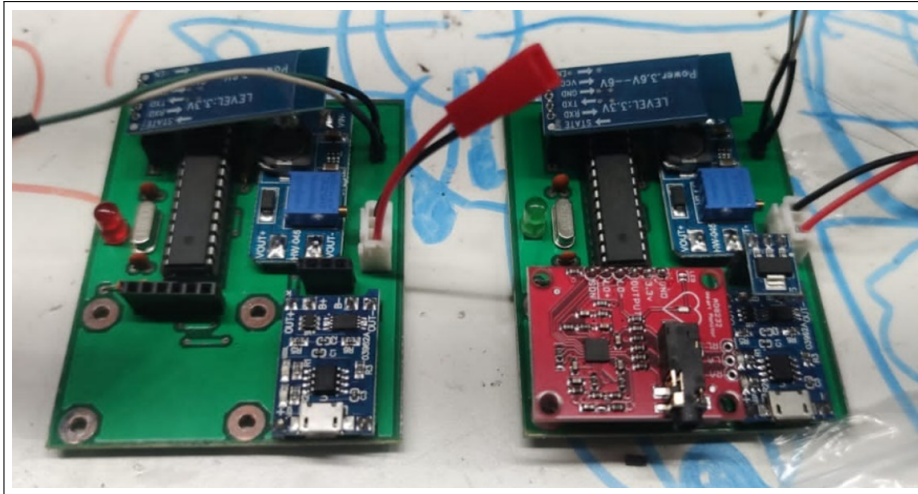
- La segunda parte es la tapa que protege el prototipo completo, en la Ilustración 4-17 se define las medidas necesarias.



**Ilustración 4-17:** Mediciones de la tapa de la carcasa  
Realizado por: Guillermo P., 2024

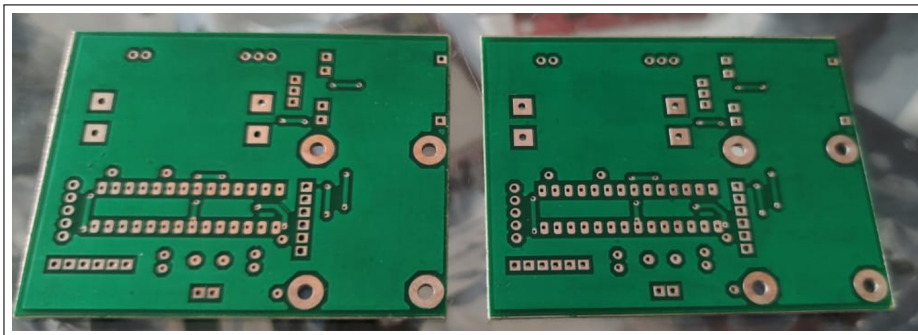
### 4.3 Etapa 3: Montaje del prototipo

La etapa final es el montaje del prototipo con todos los componentes definidos con anterioridad, en la Ilustración 4-18, se visualiza el resultado de la impresión de las placas con los componentes necesarios, vista superior, y en la Ilustración 4-19 se visualiza la impresión en la vista inferior.



**Ilustración 4-18:** Placa impresa vista superior

Realizado por: Guillermo P., 2024

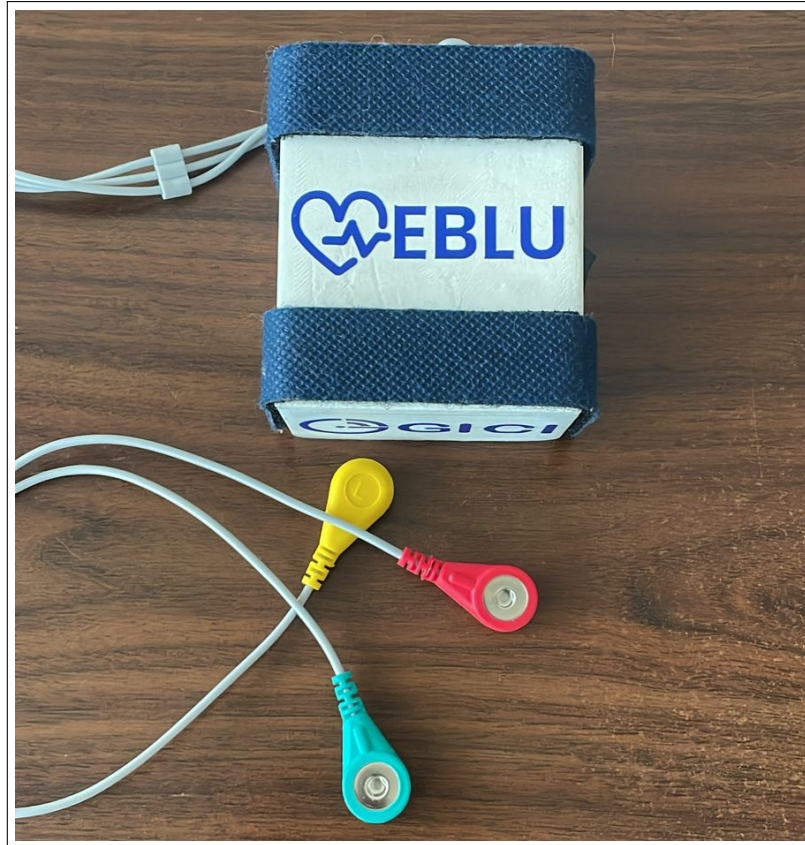


**Ilustración 4-19:** Placa impresa vista inferior

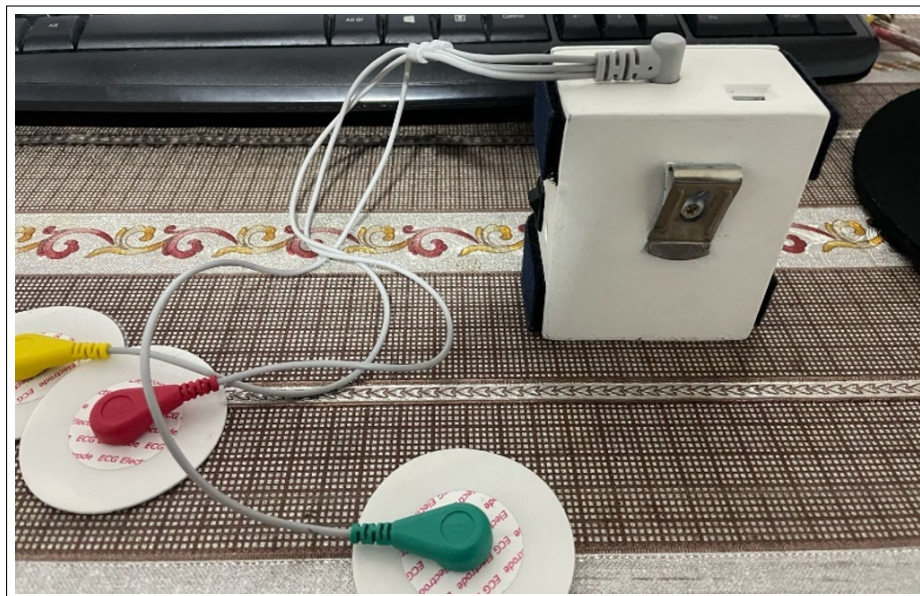
Realizado por: Guillermo P., 2024

Una vez que se verifica el funcionamiento de las placas, se procede al montaje final del prototipo, conectando los electrodos y las protecciones necesarias, en la Ilustración 4-20, se encuentra el prototipo vista frontalmente, señalando en la parte superior las conexiones de los electrodos, en la Ilustración 4-21, se encuentra el prototipo vista posterior, con una vincha que ayuda al soporte del prototipo, permitiendo la movilidad del mismo.





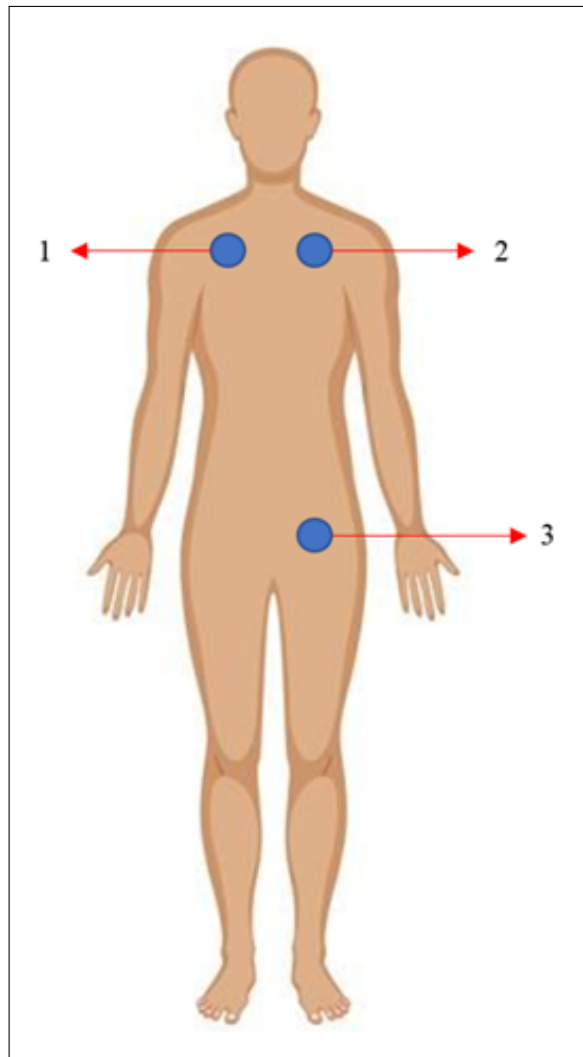
**Ilustración 4-20:** Montaje del prototipo (vista frontal)  
Realizado por: Guillermo P., 2024



**Ilustración 4-21:** Montaje del prototipo (vista posterior)  
Realizado por: Guillermo P., 2024

Como objetivo del trabajo se realiza el monitoreo con 3 electrodos, y en la Ilustración 4-22

se especifica la ubicación de los electrodos en el cuerpo del paciente, la derivación 1, se coloca debajo de la clavícula, cerca del hombro derecho, la derivación 2 se coloca debajo de la clavícula cerca del hombro izquierdo, y la derivación 3 se coloca en la parte inferior del abdomen, en cuanto a autonomía del prototipo este puede estar en funcionamiento constante por 23.5 horas gracias a su batería lipo recargable de 1000 mAh.



**Ilustración 4-22:** Ubicación de los electrodos  
Realizado por: Guillermo P., 2024

#### 4.4 Etapa 4: Validación del prototipo

Una vez realizadas las pruebas necesarias del funcionamiento, y de la recopilación de mediciones a dos pacientes, se procede a la validación del prototipo que consiste en dar a conocer a un médico especialista en el tema, en este caso un cardiólogo, con la finalidad de analizar los resultados obtenidos, tanto de las mediciones del prototipo como mediciones realizadas con un electrocardiógrafo de mercado, para ello se procede a la elaboración de



una serie de preguntas descritas en el ANEXO D: ENTREVISTA AL MÉDICO, el médico encargado de la validación del prototipo es la Dra. Sara Quizhpilema quien es residente de cardiología de segundo año del Instituto de Cardiología y Cirugía Cardiovascular.

#### 4.5 VALIDACIÓN DE PROTOTIPO

En el presente Capítulo se realiza pruebas de funcionamiento y la recolección de datos, cuya finalidad tiene la comparación del funcionamiento correcto con un electrocardiógrafo de mercado, dentro de la validación, trata de cumplir ciertos estándares y que satisfaga las necesidades y expectativas en relación a los objetivos del trabajo de titulación.

Para la comparativa de las mediciones se considera el resultado de dos personas, de diferentes edades, en donde se considera diferentes aspectos definidos a continuación:

- Tiempo de muestreo
- Monitoreo con 3 electrodos

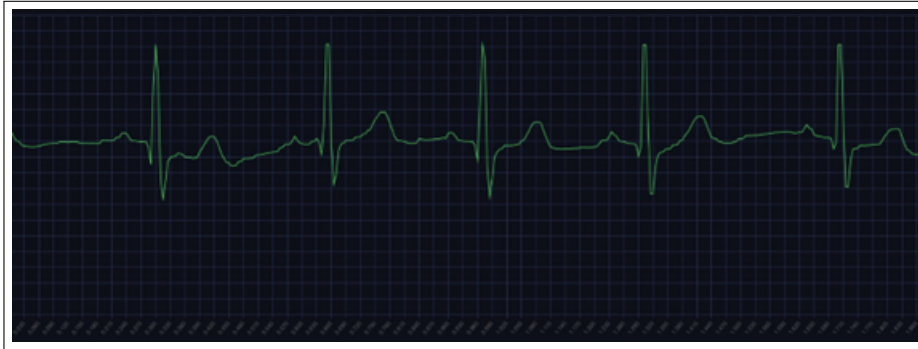
#### 4.6 Recopilación de mediciones con el prototipo

La primera persona llamada Pedro Luis Pinguil Duchi, con CI. 030232666-5 , a quien denominaremos Paciente 1, y en la Ilustración 4-23 se visualiza a la paciente 1 conectada con los electrodos del prototipo, considerando los diferentes aspectos definidos con anterioridad.



**Ilustración 4-23: Paciente 1**  
Realizado por: Guillermo P., 2024

En la Ilustración 4-24 se visualiza los resultados obtenidos del paciente 1 enviados desde la aplicación móvil a la página web.



**Ilustración 4-24:** : Resultados del Paciente 1 (página web)

Realizado por: Guillermo P., 2024

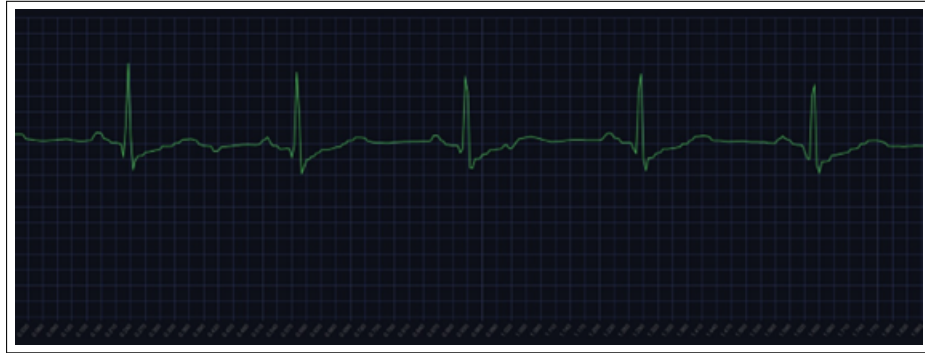
La segunda persona llamada María Encarnación Duchi Guamán, con CI. 0302743125 , a quien denominaremos Paciente 2, y en la Ilustración 4-25 se visualiza a la paciente 2 conectada con los electrodos del prototipo, considerando los diferentes aspectos definidos con anterioridad.



**Ilustración 4-25:** : Paciente 2

Realizado por: Guillermo P., 2024

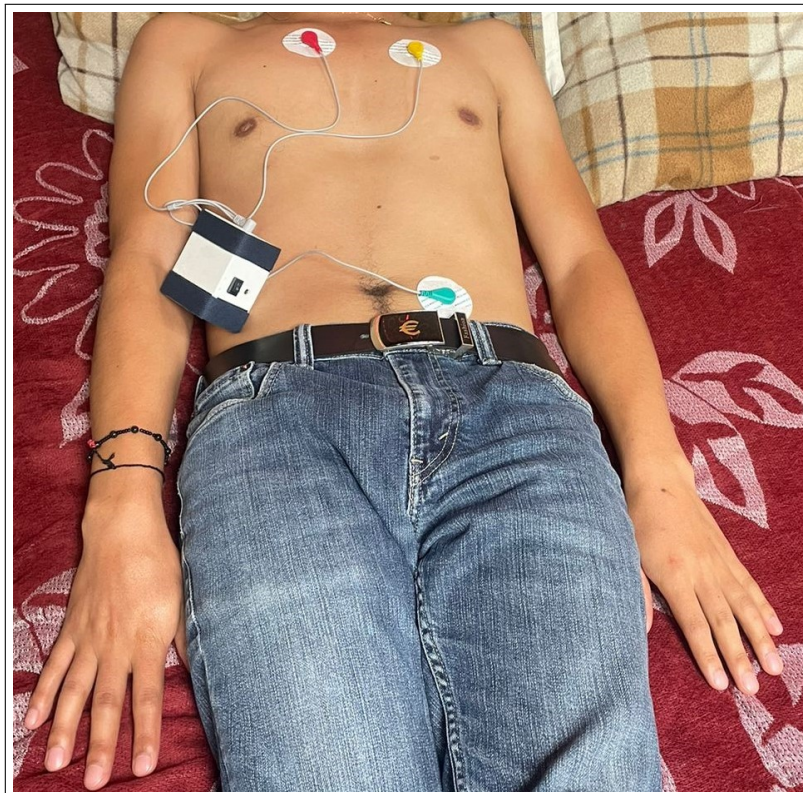
En la Ilustración 4-26 se visualiza los resultados obtenidos del paciente 2 enviados desde la aplicación móvil a la página web.



**Ilustración 4-26:** Resultados del Paciente 2 (página web)

Realizado por: Guillermo P., 2024

La tercera persona llamada Cristina Luis Sotamba Bermejo, con CI. 030280820-9 , a quien denominaremos Paciente 3, y en la Ilustración 4-27 se visualiza al paciente 3 conectada con los electrodos del prototipo, considerando los diferentes aspectos definidos con anterioridad.



**Ilustración 4-27:** Paciente 3

Realizado por: Guillermo P., 2024



En la Ilustración 4-28 se visualiza los resultados obtenidos del paciente 3 enviados desde la aplicación móvil a la página web.



**Ilustración 4-28:** Resultados del Paciente 3 (página web)

Realizado por: Guillermo P., 2024

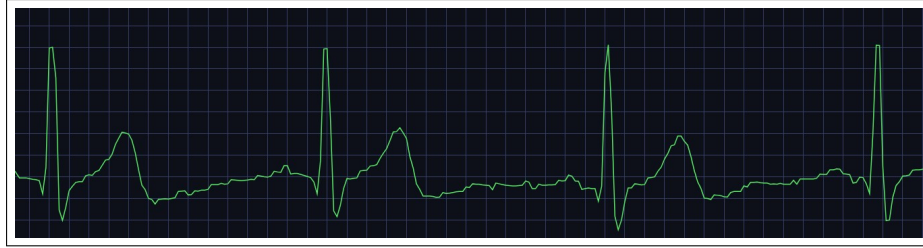
La cuarta persona llamada Juan Pablo Loja Duchi, con CI. 030265086-6 , a quien denominaremos Paciente 4, y en la Ilustración 4-29 se visualiza al paciente 4 conectada con los electrodos del prototipo, considerando los diferentes aspectos definidos con anterioridad.



**Ilustración 4-29:** Paciente 4

Realizado por: Guillermo P., 2024

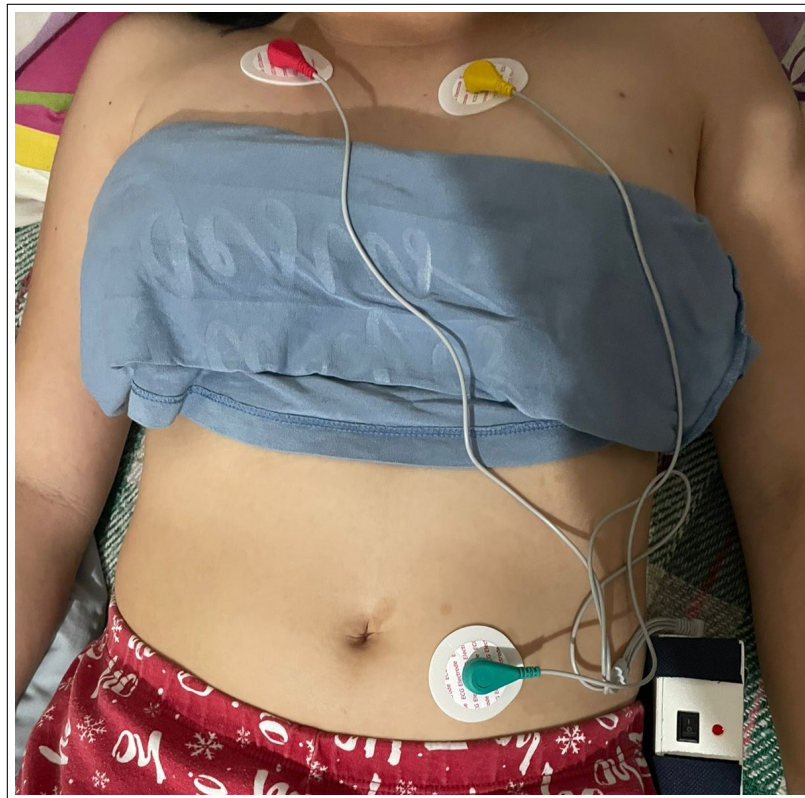
En la Ilustración 4-30 se visualiza los resultados obtenidos del paciente 4 enviados desde la aplicación móvil a la página web.



**Ilustración 4-30:** Resultados del Paciente 4 (página web)

Realizado por: Guillermo P., 2024

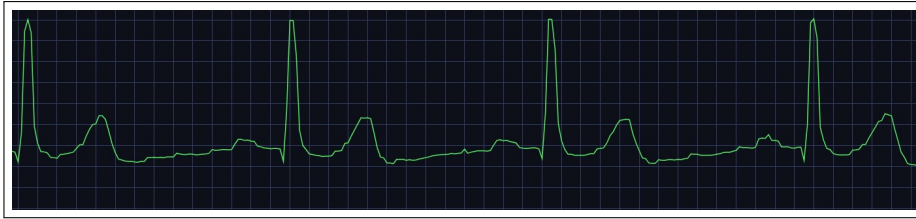
La quinta persona llamada Maria Jose Delgado Gaibor, con CI. 220019682-8 , a quien denominaremos Paciente 3, y en la Ilustración 4-31 se visualiza al paciente 3 conectada con los electrodos del prototipo, considerando los diferentes aspectos definidos con anterioridad.



**Ilustración 4-31:** Paciente 5

Realizado por: Guillermo P., 2024

En la Ilustración 4-32 se visualiza los resultados obtenidos del paciente 5 enviados desde la aplicación móvil a la página web.



**Ilustración 4-32:** Resultados del Paciente 5 (página web)

Realizado por: Guillermo P., 2024

#### 4.7 Recopilación de mediciones con un electrocardiógrafo de mercado

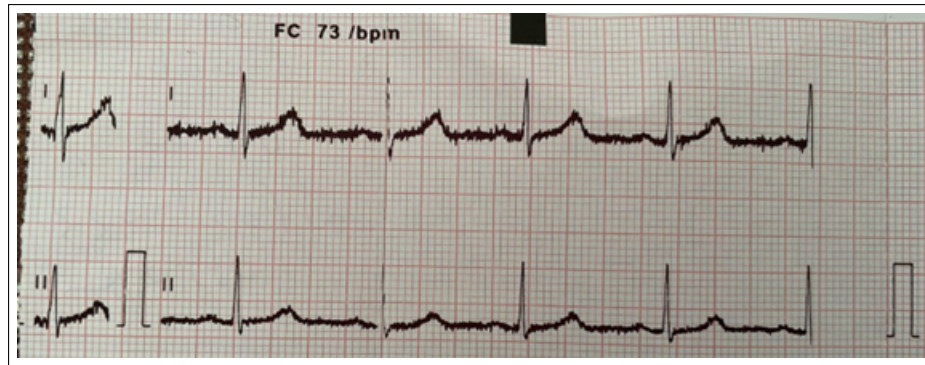
Se requiere de las mediciones realizadas por un electrocardiógrafo de mercado considerando a los mismo pacientes definidos con anterioridad, y en las mismas circunstancias, es decir el mismo tiempo de muestreo y las derivaciones de monitoreo de 3 electrodos, considerando el mismo equipo de medición en ambos, en la Ilustración 4-33 se menciona el equipo que se utilizó para las mediciones.



**Ilustración 4-33:** Mediciones del Paciente 1

Realizado por: Guillermo P., 2024

En la Ilustración 4-33 se evidencia los resultados de las mediciones realizadas en el Paciente 1, mientras que en la Ilustración 4-34 se observa los resultados de las mediciones realizadas en el Paciente 2.



**Ilustración 4-34:** : Mediciones del Paciente 2  
 Realizado por: Guillermo P., 2024

## 4.8 Comparativa del prototipo con el electrocardiógrafo

Para punto de comparación de los resultados obtenidos se tomó en consideración diferentes parámetros:

- Comparativa de los intervalos PR y QT, de los resultados del prototipo y del electrocardiógrafo de mercado.
- Latencia de la transmisión de la información desde el prototipo (módulo Bluetooth) a la aplicación móvil.
- Latencia de la transmisión de la información desde la aplicación móvil a la página web considerando el modo de conexión, ya sea por LTE o WiFi.

### 4.8.1 Comparativa del prototipo con el electrocardiógrafo

Como análisis comparativo de los resultados obtenidos tanto con el prototipo y el electrocardiógrafo de mercado, se consideró la opinión de un médico, es decir que el médico analiza las dos muestras de ambos pacientes y da a conocer las diferencias y similitudes, en la Ilustración 4-35, Ilustración 4-36, Ilustración 4-37, Ilustración 4-38 y Ilustración 4-39 se describen los resultados obtenidos



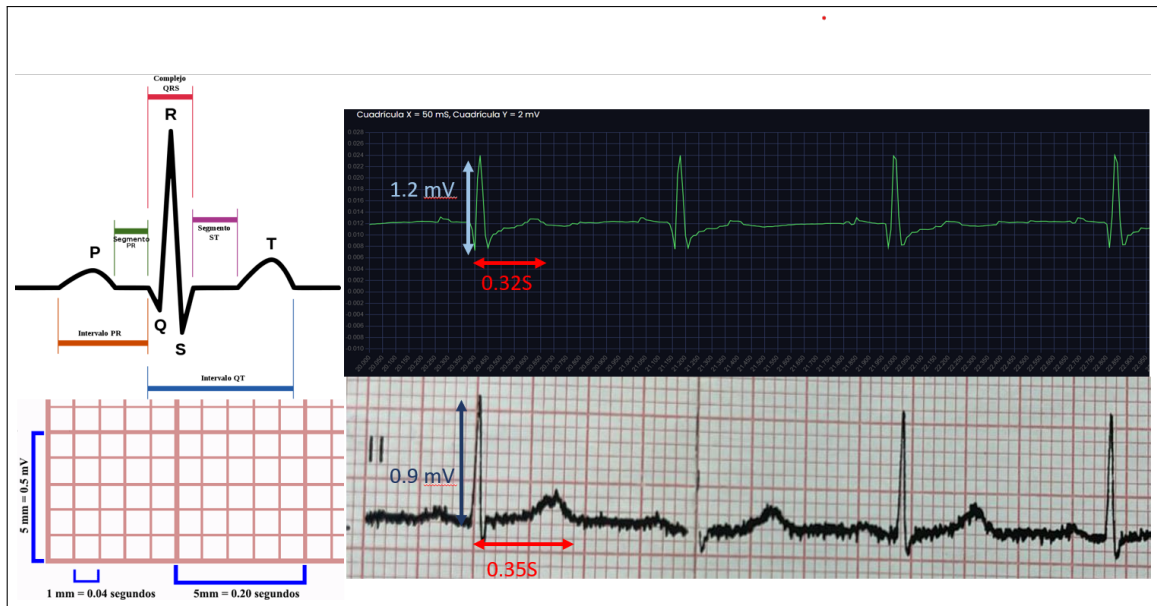


#### Ilustración 4-35: Comparación de mediciones paciente 1

Realizado por: Guillermo P., 2024

Para el paciente 1 en la gráfica de la ilustración 4-35 se observan los intervalos QR y QT. Estos datos son similares, tanto en tiempo representado en el eje X dando como resultado en el intervalo QT para el prototipo 0.33 s y en el cardiógrafo comercial 0.34 s, presentando una variación de 0.01 s. En el eje Y se describe el micro voltaje producido por los movimientos cardíacos, tomando en cuenta los puntos QR. En el prototipo se obtiene un valor de 0.7 mV y en el cardiógrafo comercial un valor de 0.8 mV, dando una variación de 0.1 mV. La gráfica en general presenta similitud, esto significa que el prototipo logra recabar la información con precisión en relación al equipo profesional, sin embargo se muestran pequeñas variaciones en la gráfica de color verde, esto debido a que el prototipo de electrocardiógrafo usa un filtrado digital y un índice de ensuavamiento de la señal. Este paciente tiene un comportamiento normal tal como describe el prototipo y el cardiógrafo industrial.





**Ilustración 4-36:** Comparación de mediciones paciente 2

Realizado por: Guillermo P., 2024

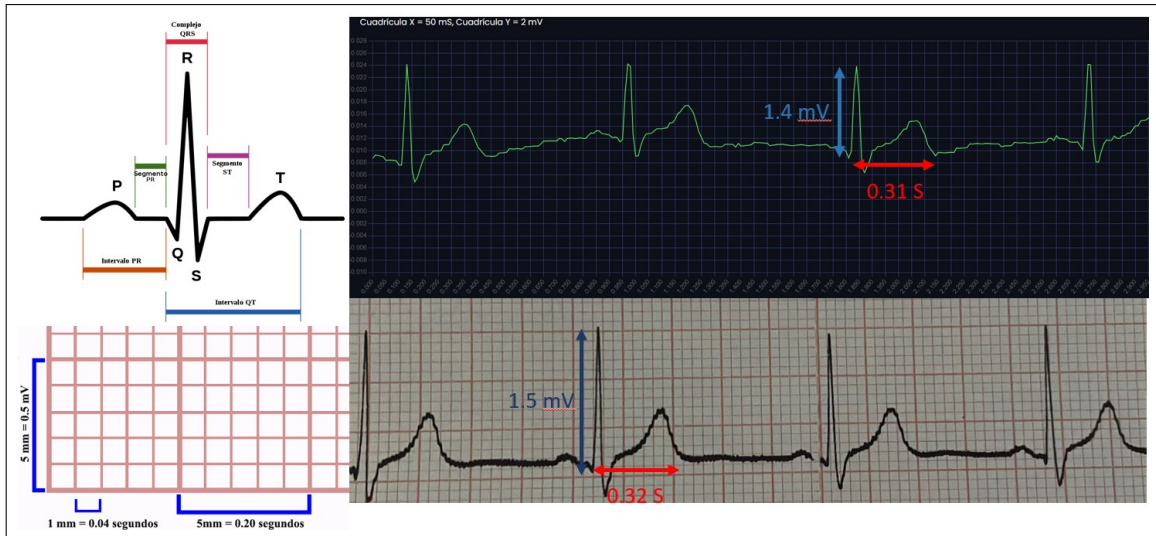
Para el paciente 2 en la gráfica de la ilustración 4-36 se observa los puntos QR y QT estos datos son similares, tanto en tiempo representado en el eje X dando como resultado en el intervalo QT para el prototipo 0.32s y en el cardiografo comercial 0.35s presentando una variacion de 0.03s y en el eje Y se describe el micro voltaje producido por los movimientos cardiacos, tomando en cuenta los puntos QR en el prototipo se obtiene un valor de 1.2mV y en el cardiografo comercial un valor de 0.9mV dando una variacion de 3mV, en esta grafica se observa que el prototipo logra mitigar de mejor manera el ruido, también describe graficas muy similares entre la gráfica obtenida con el prototipo y el equipo profesional, en la grafica se muestra unas variaciones en forma de gradas debido al procesamiento q esta realizando el prototipo para mitigar el ruido generado.



**Ilustración 4-37:** Comparación de mediciones paciente 3

Realizado por: Guillermo P., 2024

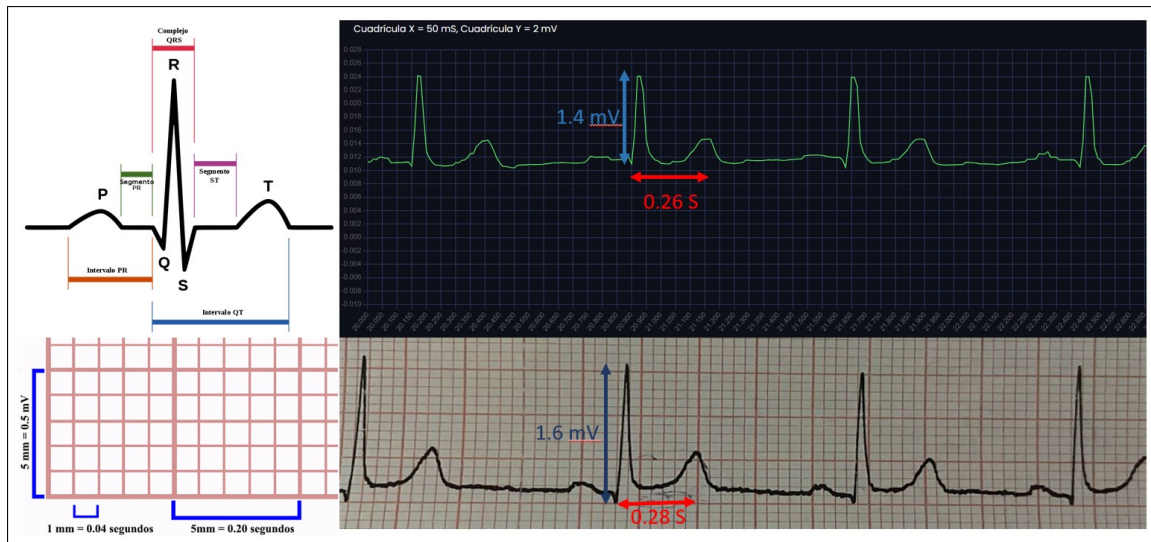
Para el paciente 3 en la gráfica de la ilustración 4-37 se observan los intervalos QR y QT, estos datos son similares, tanto en tiempo representado en el eje X dando como resultado en el intervalo QT para el prototipo 0.32 s y en el cardiografo comercial 0.33 s presentando una variación de 0.01 s y en el eje Y se describe el micro voltaje producido por los movimientos cardiacos, tomando en cuenta los puntos QR en el prototipo se obtiene un valor de 1.5mV y en el cardiografo comercial un valor de 1.6mV dando una variación de 0.1mV, en esta grafica se observa que la señal de color verde sufre cambios marcados en las curvas, esto se debe al tratamiento que se le da a la señal para mitigar el ruido sin embargo las 2 señales presentan un comportamiento similar tanto en tiempo como en potencia.



**Ilustración 4-38:** Comparación de mediciones paciente 4

Realizado por: Guillermo P., 2024

Para el paciente 4 en la gráfica de la ilustración 4-38 se observan los intervalos QR y QT. Estos datos son similares, tanto en tiempo representado en el eje X dando como resultado en el intervalo QT para el prototipo 0.31 s y en el cardiografo comercial 0.32 s presentando una variación de 0.01 s y en el eje Y se describe el micro voltaje producido por los movimientos cardiacos, tomando en cuenta los puntos QR en el prototipo se obtiene un valor de 1.4mV y en el cardiografo comercial un valor de 1.5mV dando una variación de 0.1mV, las señales en este caso muestran variaciones de la una con la otra, las curvas en cuanto a voltaje son bastante similares, la grafica verde sufre cambios debido al ruido que se genera al tomar la señal, finalmente esta señal describe un comportamiento bastante tipico del corazon sano.



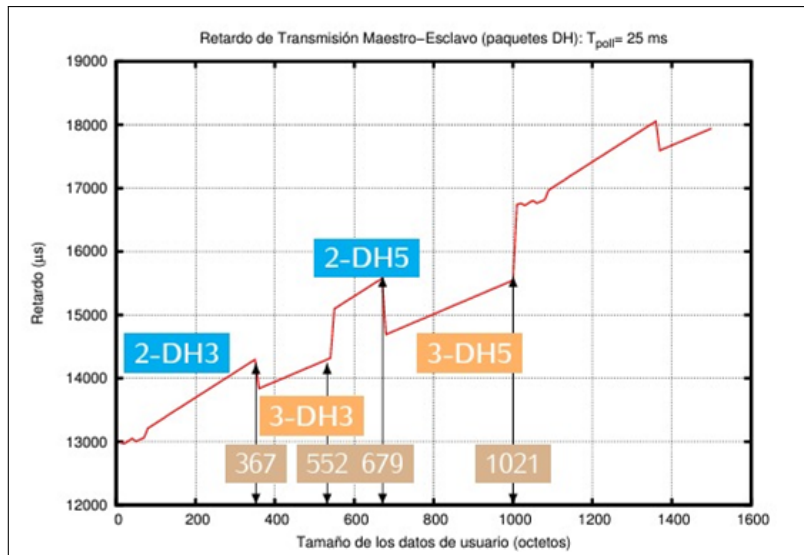
**Ilustración 4-39:** Comparación de mediciones paciente 5

Realizado por: Guillermo P., 2024

Para el paciente 5 en la gráfica de la ilustración 4-39 se observan los intervalos QR y QT, estos datos son similares, tanto en tiempo representado en el eje X dando como resultado en el intervalo QT para el prototipo 0.26 s y en el cardiografo comercial 0.28 s presentando una variación de 0.02 s y en el eje Y se describe el micro voltaje producido por los movimientos cardiacos, tomando en cuenta los puntos QR en el prototipo se obtiene un valor de 1.4mV y en el cardiografo comercial un valor de 1.6mV dando una variación de 0.2mV, la señal tomada con el prototipo se muestra estable y las curvas están bien definidas, se observa que se pierden algunos datos esto debido a que el prototipo saca una media de cada 10 datos, esto hace que se vea afectada la señal obtenida con el prototipo, en este examen se observa un comportamiento diferente, este describe una revolución temprana del corazón.

#### **4.8.2 Latencia de la transmisión de la información desde el prototipo (módulo Bluetooth) a la aplicación móvil**

La latencia de la transmisión de la información por Bluetooth es considerada de acuerdo con la información teórica disponible. Para ello, se considera la versión de Bluetooth que se utiliza, la cual es la versión 2.0, y el puerto de transmisión. El análisis de la Ilustración 4-40 define que el tiempo de retardo es de 25 ms, considerando el tamaño de los datos del usuario en octetos.



**Ilustración 4-40:** Resultados de latencia (Bluetooth)

Fuente: (GIRALDEZ, 2010)

#### 4.8.3 Latencia de la transmisión de la información desde la aplicación móvil a la página web considerando el modo de conexión, ya sea por LTE o WiFi

Para la obtención de la latencia de la transmisión de los datos desde la aplicación móvil a la página web se utiliza una aplicación extra denominada PingTool, cuenta con beneficios como Ping, Traceroute, escanear de Wi-Fi, y latencia, es una aplicación compatible con Android.

Los resultados obtenidos por la aplicación PingTool para el envío de la información desde la aplicación móvil a la página web utilizando LTE varían entre 150 ms y 263 ms. En la Ilustración 4-41 se describen los resultados de la latencia.

Ping	
www.google.com	PING
Ping www.google.com ICMP	
Desde mia09s26-in-f4.1e100.net Secuencia: 1, Tamaño: 64 bytes, TTL: 55	263 ms
Desde mia09s26-in-f4.1e100.net Secuencia: 2, Tamaño: 64 bytes, TTL: 55	150 ms
Desde mia09s26-in-f4.1e100.net Secuencia: 3, Tamaño: 64 bytes, TTL: 55	245 ms
<b>Estadísticas de ping:</b> 3 transmitido, 3 recibido, 0% paquetes perdidos, tiempo 3808 ms	
<b>Estadísticas de tiempo:</b> Min 150 \ prom 219 \ max 263 \ mdev 60.7 ms	

**Ilustración 4-41:** Resultados de latencia (LTE)

**Realizado por:** Guillermo P., 2024

Los resultados obtenidos por la aplicación PingTool para el envío de la información desde la aplicación móvil a la página web utilizando Wi-Fi varían entre 40 ms y 131 ms. En la Ilustración 4-42 se describen los resultados de la latencia.

Ping	
www.google.com	PING
Ping www.google.com 142.250.78.164 ICMP	
Desde bog02s19-in-f4.1e100.net Secuencia: 1, Tamaño: 64 bytes, TTL: 117	40 ms
Desde bog02s19-in-f4.1e100.net Secuencia: 2, Tamaño: 64 bytes, TTL: 117	83 ms
Desde bog02s19-in-f4.1e100.net Secuencia: 3, Tamaño: 64 bytes, TTL: 117	131 ms
Estadísticas de ping: 3 transmitido, 3 recibido, 0% paquetes perdidos, tiempo 3317 ms	
Estadísticas de tiempo: Min 40 \ prom 84 \ max 131 \ mdev 45.5 ms	

**Ilustración 4-42:** Resultados de latencia (Wi-Fi)

Realizado por: Guillermo P., 2024

La latencia de la transmisión de LTE en relación a la latencia por Wi-Fi es mayor, considerando que la mejor forma de envío de la información es por Wi-Fi.

#### 4.9 Análisis económico

En la Tabla?? se describe un listado de los componentes y sus costos, tanto valor unitario como valor total, y costos de los planes de la utilización de la nube, para posteriormente dar un informe total de la inversión realizada, en cuanto al prototipo, el mayor costo es la elaboración de la placa PCB, con un valor de \$37.50, con una fuente lipo de \$12.50, y los módulos principales HC-06 y AD8232 de \$6.95 y \$14.99 respectivamente, con un total de \$87.14 considerando componentes auxiliares. En cuanto a los valores de los planes de la nube depende la capacidad de memoria y de los operadores de carga, dando un valor de \$0.20, por lo que la inversión total del proyecto es de \$87.34.



**Tabla 4-1: Análisis económico**

Análisis económico			
dispositivo	cantidad	Costo Unitario	Costo Total
Placa PCB	1	\$37.50	\$37.50
Módulo Bluetooth HC-06	1	\$6.95	\$6.95
Módulo monitor de pulso cardíaco AD8232	1	\$14.99	\$14.99
Fuente de alimentación (Baterías lipo)	1	\$12.50	\$12.50
interruptor deslizante	1	\$0.25	\$0.25
electrodos	3	\$0.30	\$0.90
Cables de conexion	1	\$0.05	\$0.05
Carcasa	1	\$8.00	\$8.00
Elevador de voltaje	1	\$3.00	\$3.00
Controlador de carga	1	\$3.00	\$3.00
Costo total del prototipo			\$87.14
Costos de servicios en la nube (Firebase)			
Servicio nube (GB almacenados)	1	\$0,026 por GB	\$0.026
Servicio nube (GB descargados)	1	\$0.12 por GB	\$0.12
Servicio nube (Operaciones de carga)	1	\$0.05 por cada 10,000	\$0.05
Servicio nube (Operaciones de descarga)	1	\$0,004 por cada 10.000	\$0.004
TOTAL			\$0.2

Realizado por: Guillermo P., 2024



## CAPÍTULO V

### 5. CONCLUSIONES Y RECOMENDACIONES

#### 5.1 Conclusiones

La investigación realizada determina que el sensor cardiaco AD8232 es idóneo para la implementación del proyecto, este es un sensor utiliza electrodos para detectar los pulsos eléctricos del corazón, el mismo cuenta con características como filtrado de ruido y una ganancia de 100 veces haciendo que los valores obtenidos puedan ser fácilmente diferenciados y analizados.

Para el desarrollo del prototipo del cardiógrafo se utilizó un software de diseño de circuitos eléctricos en el cual se hizo la simulación del funcionamiento de todos los módulos en conjunto dando resultados favorables en cuanto al funcionamiento del sistema en conjunto, este tiene la capacidad de obtener los voltajes producidos por el corazón y enviar estos datos utilizando tecnología bluetooth.

En la implementación del cardiógrafo móvil se utilizó una placa PCB impresa de manera industrial la cual está encargada de conectar todos los módulos para que estos funcionen como un sistema, se obtuvo buenos resultados en cuanto a la capacidad de muestreo los datos, enviando datos tenemos resultados de tiempos de retardo de 85.7 ms con redes móviles y 60.7 ms utilizando una red wifi en cuanto a autonomía del sistema se obtiene un promedio de duración de la batería de 23.5 horas.

La aplicación para el smartphone Android cuenta con la capacidad de recibir los datos enviados desde el cardiógrafo usando la tecnología bluetooth, graficar estos datos en tiempo real, crear contenedores en la base de datos utilizando los datos del paciente para guardar los datos obtenidos y finalmente enviar los datos a los contenedores ya definidos permitiendo una comunicación en tiempo real.

La evaluación de funcionamiento del prototipo mediante la comparación con un electrocardiógrafo comercial da como resultados variaciones mínimas, analizando el eje Y definida para valores de potencia definidos en mv tenemos variaciones máximas de 0.3 mv en cuanto al análisis del eje X el cual es la representación del tiempo tenemos variaciones máximas de 0.03 s, concluyendo que el prototipo obtiene datos muy similares en cuanto a potencia, tiempos y forma de la gráfica obtenida.

## 5.2 Recomendaciones

Seria factible poder usar un sensor que permita recolectar una mayor cantidad de derivaciones cardiacas para obtener un estudio más amplio de todo el comportamiento eléctrico del corazón, esto brindaría más información para que el medico tratante pueda diagnosticar posibles complicaciones.

Se podría mejorar el diseño del prototipo para que este sea más compacto dándole mas facilidad para la portabilidad, en cuanto a la autonomía se podría mejorar la batería para brindarle mayor autonomía al prototipo.

Para que el prototipo de mediciones mas precisas es importante que este sea usado en ambientes de bajo ruido y el paciente estando lo mas estático posible, esto le brindara mayor precisión y evitara que el ruido haga cambios en las señales medidas.

## BIBLIOGRAFÍA

**AHAMMED, S. & PILLAI, B. C.** *Design of Wi-Fi based mobile electrocardiogram monitoring system on Concerto platform. Procedia Engineering* [en línea], 2013, () vol. 64, págs. 65-73 Disponible en: <https://doi.org/10.1016/j.proeng.2013.09.077>.

**ANATOMIA DEL CORAZON.** <https://www.texasheart.org/heart-health/heart-information-center/topics/anatomia-del-corazon/>. [En línea] Citado el 19 de noviembre de 2023.

**AV ELECTRONICS, Sensor Pulso.** 2023-11. <https://avelectronics.cc/producto/sensor-pulso/>.

**AXARNET,** Tipos de plataformas para desarrollar apps. Plataformas de desarrollo de aplicaciones móviles. 2023-agosto 4. Recuperado el 11 de noviembre de 2023, de <https://axarnet.es/blog/plataformas-desarrollo-aplicaciones-moviles#>.

**BENIGNO, Jacob:** DigiKey. <https://www.digikey.com/es/articles/how-to-select-and-use-the-right-esp32-wi-fi-bluetooth-module>. [En línea] 21 de enero de 2020. [Citado el: 20 de noviembre de 2023].

**BIKEPIXELS,** How to Add a LiPo Battery in Our Arduino Projects. 2019-11. <https://bikepixels.com/2019/11/10/how-to-add-a-lipo-battery-in-our-arduino-projects/>.

**ELECTROCARDIÓGRAFOS.** [https://enventa.ec/electrocardiografos/#Electrocardiografos\\_multicanal](https://enventa.ec/electrocardiografos/#Electrocardiografos_multicanal). [En línea] 5 de octubre de 2022.

**FERNANDEZ, Yúbal:** Xataka - Tecnología y gadgets, móviles, informática, electrónica. <https://www.xataka.com/basics/bluetooth-diferencias-caracteristicas-sus-clases-versiones>. [En línea] 21 de abril de 2020. [Citado el: 18 de noviembre de 2023].

**FORERO, María:** hctmrussi. <https://hctmrussi.wordpress.com/2018/05/24/electrocardiografo-maria-alejandra-villamil-forero/>. [En línea] 20 de mayo de 2018. [Citado el: 6 de noviembre de 2023].

**GARCIA, Juan Pascual, & MOLINA, José María, & GARCIA-PADRON, & LLACER, Leandro Juan:** Sistema de comunicaciones móviles. <https://repositorio.upct.es/bitstream/handle/10317/8854/9788494294464.pdf?sequence=1&isAllowed=y>. [En línea] 2014. [Citado el: 19 de noviembre de 2023].

**GIRALDEZ, Jose:** Curva del retardo de transmisión para el perfil SPP en Bluetooth 2.0 + EDR. [chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/http://webpersonal.uma.es/~ECASILARI/Research/Tesis/04Presentacion-Pub.pdf](http://webpersonal.uma.es/~ECASILARI/Research/Tesis/04Presentacion-Pub.pdf). Retardo bluetooth.

**GISI.** <https://virtual.cuautitlan.unam.mx/intar/sistdig/tarjetas-de-desarrollo/>. [En línea] SF de SF de SF. [Citado el: 20 de noviembre de 2023].

**IONOS,** Digital Guide IONOS. <https://www.ionos.es/digitalguide/servidores/know-how/que-es-bluetooth/>. [En línea] 1 de diciembre de 2022. [Citado el: 19 de noviembre de 2023].

**KALSTEIN** <https://kalstein.net/es/que-es-un-electrocardiografo-monocanal/>. [En línea] 4 de junio de 2022.

**LEMUS, Isaac:** Conocimiento libre. <https://conocimientolibre.mx/microcontrolador/>. [En línea] 20 de agosto de 2019. [Citado el: 20 de noviembre de 2023].

**MATUTE, Alex, & REINO, Mario,** Diseño e implementación de un electrocardiógrafo ambulatorio dinámico de una derivación con conexión inalámbrica a teléfono móvil inteligente y envío de información vía SMS o GPRS(), 2012.

**MEDIOS-MEDICOS-DEL-NUCLEO,** Corazón norma [Fuente de la imagen]. <https://catalog.nucleusmedicalmedia.com/coraz%C3%B3n-normal/view-item?ItemID=26960>.

**MEGATRONICA.**<https://megatronica.cc/producto/modulo-bluiethoot-hc-06/>. [En línea] SF de SF de SF. [Citado el: 18 de noviembre de 2023].

**NATIONAL GEOGRAPHIC.** [https://www.nationalgeographic.com.es/ciencia/que-es-5g-y-como-nos-cambiara-vida\\_14449](https://www.nationalgeographic.com.es/ciencia/que-es-5g-y-como-nos-cambiara-vida_14449). [En línea] 15 de diciembre de 2022. [Citado el: 19 de noviembre de 2023].

**NOVATRONIC** <https://novatronicec.com/index.php/product/ad8232-ecg-modulo-monitor-de-pulso-cardiaco/>. [En línea] 2023. [Citado el: 19 de noviembre de 2023].

**PEREZ, F. SN.** <http://www.tierradelazaro.com/wp-content/uploads/2016/04/4G.pdf>. [En línea] 2010. [Citado el: 19 de noviembre de 2023].

**RED-HAT,** Red Hat. <https://www.redhat.com/es/topics/cloud-computing/what-are-cloud-services>. [En línea] 10 de julio de 2023. [Citado el: 20 de noviembre de 2023].

**ROBOTS-ARGENTINA,** Convertidor de Voltaje MT3608 [<https://robots-argentina.com.ar/didactica/convertidor-de-voltaje-mt3608/>].

**RPEREXP,** Gobiernodecanarias. <https://www3.gobiernodecanarias.org/medusa/ecoblog/rperexp/2022/03/29/app-inventor-2022/>. [En línea] 29 de marzo de 2022. [Citado el: 20 de noviembre de 2023].

**RUIZ, A.** ticnegocios. <https://app.bibguru.com/p/3c14b5e1-184f-4c82-9768-8fba2a9964ac>. [En línea] 18 de Julio de 2018.

**SALAZAR, Jordi:** Clasificación Redes Inalámbricas. [chrome-extension://efaidnbmnnnibpcajpcg1cl1efindmkaj/https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01\\_R\\_ES.pdf](chrome-extension://efaidnbmnnnibpcajpcg1cl1efindmkaj/https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf). Redes Inalámbricas.

**SALAZAR, Jordi:** Mockup-Driven. [https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01\\_R\\_ES.pdf](https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf). Tesis de maestría, Universidad Politécnica de Cataluña.

**SALAZAR, Jordi:** Tech pedia. [https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01\\_R\\_ES.pdf](https://upcommons.upc.edu/bitstream/handle/2117/100918/LM01_R_ES.pdf). [En línea] SF de SF de SF. [Citado el: 20 de noviembre de 2023].

**SANTAELLA, Jesús:** Talently. <https://talently.tech/blog/que-es-android-studio/>. [En línea] 26 de abril de 2022. [Citado el: 20 de noviembre de 2023].

**SMITH, Rafael, & COBO, Rubén, & VAZQUES, Carlos.** *Diseño de un sistema inalámbrico de monitorización electrocardiográfica para dispositivos Android.* *scielo* [en línea], 2020, (20 de Noviembre de 2023) vol. 4 Disponible en: [http://scielo.sld.cu/scielo.php?script=sci\\_arttext&pid=S1815-59282020000200063](http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1815-59282020000200063).

**VARGAS, Laura:** Arduino uno [Fuente de la imagen]. <https://www.researchgate.net/profile/Laura-Vargas-7/publication/305477991/figure/fig3/AS:469333089492992@1488909136041/Arduino-UNO-Las-principales-caracteristicas-de-esta-tarjeta-es-que-cuenta-con-seis.png>.

**ZAVALA, José.** *Descripción del electrocardiograma normal y lectura del electrocardiograma.* *Revista Mexicana de anestesiología* [en línea], 2013, ( ) vol. 40 Disponible en: <chrome-extension://efaidnbmnnnibpcajpcg1cl1efindmkaj/https://www.medigraphic.com/pdfs/rma/cma-2017/cmas171bj.pdf>.

## ANEXOS

### ANEXO A: CODIGO ELECTROCARDIOGRAFO BLUETOOTH (IDE ARDUINO)

```
#include <SoftwareSerial.h>

const int analogInputPin = A0;      // Pin analógico de entrada
const int filteredOutputPin = 9;    // Pin de salida del filtro

const int numSamples = 30;         // Número de muestras para promediar
int samples[numSamples];           // Array para almacenar las muestras
int sampleIndex = 0;               // Índice de la muestra actual
int filteredValue = 0;             // Valor filtrado inicial

const float alpha = 0.7;           // Factor de suavizado (0 < alpha < 1)
const int hysteresis = 10;         // Valor de histeresis

SoftwareSerial bluetoothSerial(3, 4); // RX, TX para Bluetooth

void setup() {
  Serial.begin(9600);
  bluetoothSerial.begin(9600);
  pinMode(10, INPUT);
  pinMode(11, INPUT);
  pinMode(filteredOutputPin, OUTPUT); // Configurar el pin de salida del filtro
}

void loop() {
  static int counter = 0;

  if ((digitalRead(11) == 1) || (digitalRead(10) == 1)) {
    if (counter % 1 == 0) {
      Serial.println("%");
    }
  } else {
    if (counter % 1 == 0) {
      // Leer el valor analógico de entrada
    }
  }
}
```

```
int sensorValue = analogRead(analogInputPin);

// Aplicar histeresis
if (abs(sensorValue - filteredValue) > hysteresis) {
// Solo actualizar si la diferencia es mayor que el valor de histeresis
filteredValue = sensorValue;
}

// Almacenar la muestra en el array
samples[sampleIndex] = filteredValue;

// Actualizar el índice de la muestra
sampleIndex = (sampleIndex + 1) % numSamples;

// Calcular el promedio ponderado con un filtro de paso bajo
filteredValue = alpha * filteredValue + (1 - alpha) * sensorValue;

// Enviar el valor filtrado por el monitor serie y Bluetooth
//Serial.println(filteredValue);
bluetoothSerial.println(filteredValue);
}
}

counter++;
delay(5);
}
```

## ANEXO B: CODIGO APLICACION MÓVIL

```
package com.guille.eblu

import android.Manifest
import android.app.Activity
import android.bluetooth.BluetoothAdapter
import android.bluetooth.BluetoothDevice
import android.bluetooth.BluetoothManager
import android.bluetooth.BluetoothSocket
import android.content.Context
import android.content.Intent
import android.content.pm.PackageManager
import android.graphics.Color
import android.os.Bundle
import android.util.Log
import android.view.inputmethod.EditorInfo
import android.view.inputmethod.InputMethodManager
import android.widget.AdapterView
import android.widget.AdapterView.OnItemClickListener
import android.widget.ArrayAdapter
import android.widget.Button
import android.widget.EditText
import android.widget.Spinner
import android.widget.Switch
import android.widget.Toast
import androidx.activity.result.contract.ActivityResultContracts.StartActivityForResult
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import java.io.InputStream
import java.util.UUID
import android.widget.TextView
import androidx.activity.result.ActivityResultLauncher
import androidx.activity.result.contract.ActivityResultContracts
import com.github.mikephil.charting.charts.LineChart
import com.github.mikephil.charting.components.Description
```



```

import com.github.mikephil.charting.data.Entry
import com.github.mikephil.charting.data.LineData
import com.github.mikephil.charting.data.LineDataSet
import com.github.mikephil.charting.formatter.ValueFormatter
import com.github.mikephil.charting.interfaces.datasets.ILineDataSet
import com.google.firebase.database.FirebaseDatabase

import java.io.BufferedReader
import java.io.IOException
import java.io.InputStreamReader
import java.text.SimpleDateFormat
import java.util.Date
import java.util.LinkedList
import java.util.Locale

const val REQUEST_ENABLE_BT = 1
private val REQUEST_CODE_BLUETOOTH_PERMISSION = 2

class MainActivity : AppCompatActivity() {

    private lateinit var chart: LineChart
    private lateinit var textView: TextView
    private val lineEntries: ArrayList<Entry> = ArrayList()

    lateinit var mBtAdapter: BluetoothAdapter
    var mAddressDevices: ArrayAdapter<String>? = null
    var mNameDevices: ArrayAdapter<String>? = null
    var isUploading = false

    companion object {

        // private lateinit var textView: TextView
        var m_myUUID: UUID = UUID.fromString("00001101-0000-1000-8000-00805F9B34FB")
        private var m_bluetoothSocket: BluetoothSocket? = null

```

```

var m_isConnected: Boolean = false
lateinit var m_address: String
}
private lateinit var someActivityResultLauncher: ActivityResultLauncher<Intent>

override fun onCreate(savedInstanceState: Bundle?) {
super.onCreate(savedInstanceState)
setContentView(R.layout.activity_main)
// Inicializar someActivityResultLauncher
someActivityResultLauncher = registerForActivityResult(ActivityResult
Contracts.StartActivityForResult()) { result ->
if (result.resultCode == Activity.RESULT_OK) {
// El Bluetooth ha sido activado exitosamente
Toast.makeText(this, "Bluetooth activado", Toast.LENGTH_LONG).show()
} else {
// El usuario ha cancelado la solicitud de activación del Bluetooth
Toast.makeText(this, "Activación del Bluetooth cancelada por el usuario",
Toast.LENGTH_LONG).show()
}
}
}
//dddddddddddddddddddddddddddddddddddddddddddddddddd
mAddressDevices = ArrayAdapter(this, android.R.layout.simple_list_item_1)
mNameDevices = ArrayAdapter(this, android.R.layout.simple_list_item_1)
val idBtnOnBT = findViewById<Button>(R.id.idBtnOnBT)
val idBtnConect = findViewById<Button>(R.id.idBtnConect)
val idBtnDispBT = findViewById<Button>(R.id.idBtnDispBT)
val idSpinDisp = findViewById<Spinner>(R.id.idSpinDisp)
val switch1 = findViewById<Switch>(R.id.idSwiCloud)
val switch = findViewById<Switch>(R.id.idSwiGraph)
chart = findViewById(R.id.chartView)
textView = findViewById(R.id.dataTextView)

val someActivityResultLauncher = registerForActivityResult(
StartActivityForResult())

```

```

) { result ->
if (result.resultCode == REQUEST_ENABLE_BT) {
Log.i("MainActivity", "ACTIVIDAD REGISTRADA")
}
}

//Inicializacion del bluetooth adapter
mBtAdapter = (getSystemService
(Context.BLUETOOTH_SERVICE)
as BluetoothManager).adapter

val REQUEST_CODE_LOCATION_PERMISSION = 1001

//Boton Encender bluetooth
idBtnOnBT.setOnClickListener {
if (mBtAdapter.isEnabled) {
// Si el Bluetooth ya está activado
Toast.makeText(this, "Bluetooth ya se encuentra activado", Toast.LENGTH_LONG).show()
} else {
// Verificar si se tienen los permisos
if (ActivityCompat.checkSelfPermission(
this,
Manifest.permission.BLUETOOTH_CONNECT
) != PackageManager.PERMISSION_GRANTED ||
ActivityCompat.checkSelfPermission(
this,
Manifest.permission.ACCESS_FINE_LOCATION
) != PackageManager.PERMISSION_GRANTED
) {
// Si no se tienen los permisos, solicitarlos al usuario
ActivityCompat.requestPermissions(
this,
arrayOf(

```

```

Manifest.permission.BLUETOOTH_CONNECT,
Manifest.permission.ACCESS_FINE_LOCATION
),

val device: BluetoothDevice = mBtAdapter.getRemoteDevice(m_address)
m_bluetoothSocket = device.createInsecureRfcommSocketToServiceRecord(m_myUUID)
m_bluetoothSocket!!.connect()

Toast.makeText(this, "Conectado", Toast.LENGTH_LONG).show()

} else {
Toast.makeText(this, "Selecciona un dispositivo para conectar",
Toast.LENGTH_LONG).show()
}
} else {
Toast.makeText(this, "Ya estás conectado a un dispositivo", Toast.LENGTH_LONG).show()
}
} catch (e: Exception) {
e.printStackTrace()
Toast.makeText(this, "Error al conectar al dispositivo", Toast.LENGTH_LONG).show()
}
}

////////////////////////////////////
var isRunning = false // Variable para controlar el estado de ejecución

var isRunningUpload = false

switch.setOnCheckedChangeListener { _, isChecked ->
if (isChecked) {
isRunning = true // Inicia la ejecución del código
Thread {
// Configurar descripción del gráfico
val description = Description()
description.text = "Datos en tiempo real"
chart.description = description

```

```

// Configurar eje Y
val yAxis = chart.axisLeft
yAxis.axisMinimum = 100f
yAxis.axisMaximum = 800f
// Configurar eje X (opcional)
val startTime = System.currentTimeMillis() // Tiempo de inicio de la
recepción de datos en milisegundos

// Configurar el formateador de valores del eje x para mostrar el tiempo transcurrido
val xAxis = chart.xAxis
xAxis.isEnabled = true
xAxis.labelCount = 8 // Número de etiquetas en el eje x
xAxis.valueFormatter = object : ValueFormatter() {
    override fun getFormattedValue(value: Float): String {
        // Calcular el tiempo transcurrido desde el inicio de la recepción de
        datos en segundos
        val currentTime = System.currentTimeMillis() - startTime
        val elapsedSeconds = currentTime.toFloat() / 1000

        // Formatear el tiempo transcurrido con 2 decimales
        val formattedSeconds = String.format("%.2f", elapsedSeconds)

        // Devolver el tiempo formateado
        return "$formattedSeconds s"
    }
}

// Configurar datos y conjunto de datos
val lineDataSet = LineDataSet(lineEntries, "Datos")
lineDataSet.color = Color.BLUE
lineDataSet.setDrawCircles(false)
val lineData = LineData(lineDataSet)
chart.data = lineData
////////////////////////////////////

val inputStream: InputStream? = m_bluetoothSocket?.inputStream
val n = 3 // Define el valor de n según el número de datos no procesados
antes de procesar uno

```

```

val maxEntries = 150 // Define el límite máximo de entradas en el gráfico
val updateInterval = 5 // Define el intervalo de actualización del gráfico

val lineEntries: LinkedList<Entry> = LinkedList()
// Utiliza una LinkedList para un borrado más eficiente

try {
val reader = BufferedReader(InputStreamReader(inputStream))

for (value in values) {
synchronized(lineEntries) { // Bloque sincronizado para evitar
modificaciones concurrentes
if (dataIndex == 0) {
lineEntries.addLast(Entry(processedCounter.toFloat(), value)) // Agrega al final
de la lista
if (lineEntries.size > maxEntries) {
lineEntries.removeFirst() // Elimina el primer elemento de la lista
}
processedCounter++
}
}

dataIndex = (dataIndex + 1) % n

updateCounter++
if (updateCounter >= updateInterval) {
val textValue = values.lastOrNull()?.toString() ?: ""
runOnUiThread {
textView.text = textValue

val lineDataSets: ArrayList<ILineDataSet> = ArrayList()
synchronized(lineEntries) { // Bloque sincronizado para evitar
modificaciones concurrentes
lineDataSets.add(LineDataSet(lineEntries, "Real-time Data").apply {
color = Color.BLUE
setDrawCircles(false)
setDrawValues(false)
}
}
}
}
}

```

```

    })
}

val newData = LineData(lineDataSets)

chart.data = newData
chart.invalidate()
}
updateCounter = 0
}
}
}
} catch (e: IOException) {
    e.printStackTrace()
} finally {
    //inputStream?.close()
    //m_bluetoothSocket?.close()
}
}.start()
} else {
    isRunning = false // Detiene la ejecución del código
    lineEntries.clear()

    runOnUiThread {
        chart.clear()
        chart.invalidate()
    }
}

//*****
val editTextCedula: EditText = findViewById(R.id.editTextCedula)

switch1.setOnCheckedChangeListener { _, isChecked ->
    if (isChecked) {
        // Mostrar mensaje al inicio del proceso
        Toast.makeText(this, "Agrega el número de cédula y nombre

```

```

(separados por ,)", Toast.LENGTH_SHORT).show()

isRunningUpload = true

editTextCedula.setOnEditorActionListener { _, actionId, _ ->
if (actionId == EditorInfo.IME_ACTION_DONE) {
val inputText = editTextCedula.text.toString()
val inputData = inputText.split(",")

if (inputData.size >= 2) {
val cedula = inputData[0].trim()
val nombre = inputData[1].trim()

if (cedula.isNotEmpty() && nombre.isNotEmpty()) {
// Ocultar el teclado al presionar "Listo"
val imm = getSystemService(INPUT_METHOD_SERVICE) as InputMethodManager
imm.hideSoftInputFromWindow(editTextCedula.windowToken, 0)

// Aquí comienza la lógica de subida de datos
isUploading = true
subirNombre(cedula, nombre)
subirDatos(cedula,nombre)
} else {
Toast.makeText(this, "Por favor, ingresa la cédula y el nombre",
Toast.LENGTH_SHORT).show()
}
} else {
Toast.makeText(this, "Formato incorrecto. Ingresa cédula y
nombre separados por ,", Toast.LENGTH_SHORT).show()
}
}
true
}
} else {

isRunningUpload = false
isUploading = false

```



```

//isRunning = false // Detiene la ejecución del código
// lineEntries.clear()
// chart.clear()
// chart.invalidate()
// Mostrar mensaje al desactivar el switch
Toast.makeText(this, "Datos subidos", Toast.LENGTH_SHORT).show()
// Lógica cuando se desactiva el switch
}
}
//*****

////////////////////////////////////

}

override fun onRequestPermissionsResult(
requestCode: Int,
permissions: Array<out String>,
grantResults: IntArray
) {

fun prompt(message: String): String? {
print(message)
return readLine()
}

// Función para subir solo el nombre
private fun subirNombre(cedula: String, nombre: String) {
Thread {
val database = FirebaseDatabase.getInstance()
val databaseReference = database.reference.child(cedula)

// Usar nombre y cédula como parte de la referencia para almacenar
el nombre
val nuevaLecturaRef = databaseReference.child("nombre")
nuevaLecturaRef.setValue(nombre)

// Ahora puedes iniciar la subida de datos recibidos por Bluetooth

```

```

si es necesario
// Puedes llamar a la función subirDatosBluetooth aquí si quieres
iniciarla inmediatamente después de subir el nombre
}.start()
}

// Función modificada para subir datos con nombre, cédula y fecha
private fun subirDatos(cedula: String, nombre: String) {
Thread {
val database = FirebaseDatabase.getInstance()
val databaseReference = database.reference.child(cedula)

val inputStream: InputStream? = m_bluetoothSocket?.inputStream
val reader = BufferedReader(InputStreamReader(inputStream))

try {
// Mostrar mensaje al inicio de la subida de datos
runOnUiThread {
Toast.makeText(this, "Subiendo datos", Toast.LENGTH_SHORT).show()
}

// Subir el nombre y la fecha una sola vez
val fechaActual = obtenerFechaActual()
databaseReference.child("nombre").setValue(nombre)
databaseReference.child("fecha").setValue(fechaActual)

while (isUploading) {
val line = reader.readLine() ?: break
val values = line.split(",").mapNotNull { it.toFloatOrNull() }

val valorRecibido = values.lastOrNull() ?: 0.0f

// Usar cédula como parte de la referencia para almacenar los datos
val nuevaLecturaRef = databaseReference.child("lecturas").push()
nuevaLecturaRef.setValue(valorRecibido)
}
}
}
}

```

```
}  
} catch (e: IOException) {  
    e.printStackTrace()  
} finally {  
    // Una vez que se termina de enviar, mostrar el mensaje  
    runOnUiThread {  
        Toast.makeText(this, "Datos subidos", Toast.LENGTH_SHORT).show()  
    }  
}  
}.start()  
}  
  
// Función para obtener la fecha actual formateada  
private fun obtenerFechaActual(): String {  
    val formatoFecha = SimpleDateFormat("yyyy-MM-dd HH:mm:ss",  
        Locale.getDefault())  
    return formatoFecha.format(Date())  
}  
  
}
```

## ANEXO C: CODIGO PAGINA WEB

```
<script>  
</script>
```

```
<div class="card">  
  <slot></slot>  
</div>
```

```
<style scoped lang="scss">  
  div.card {  
    display: grid;  
    place-items: center;  
    position: relative;  
  
    padding: 0;  
    width: 18.125rem;  
    height: 18.125rem;  
  
    background-color: var(--card-back-color);  
    color: var(--text-color);  
    border-radius: 0.5rem;  
  
    &::before {  
      content: '';  
  
      position: absolute;  
      z-index: -2;  
  
      width: 18.75rem;  
      height: 18.75rem;  
      z-index: -2;  
  
      border-radius: 0.75rem;  
      background: linear-gradient(-45deg, #e81cff 0%, #40c9ff 100%);  
  
      transition: all 0.6s cubic-bezier(0.175, 0.885, 0.32, 1.275);
```

```
    }

    &::after {
      content: "";

      position: absolute;
      z-index: -1;
      inset: 0;

      background: linear-gradient(-45deg, #fc00ff 0%, #00dbde 100%);
      transform: translate3d(0, 0, 0) scale(0.95);
      filter: blur(1rem);
    }

    &:hover::after {
      filter: blur(1rem);
    }

    &:hover::before {
      transform: rotate(-90deg);
    }
  }
</style>
```

```
<script>
  import Chart from "chart.js/auto";

  import { onMount } from "svelte";

  import { signs } from "../stores";

  let isActive = false;
  let canvas;
  let chart;
  let box;
```

```
// config...
const config = {
  type: "line",
  data: {
    datasets: [
      {
        label: "Cuadrícula X = 5 mS, Cuadrícula Y = 1 mV",
        borderColor: "#59ed68",
        borderWidth: 1.5,
        radius: 0,
      },
    ],
  },
  options: {
    responsive: true,
    maintainAspectRatio: false,
    showTooltips: false,
    animation: false,
    scales: {
      x: {
        type: "linear",
        min: 0,
        position: "bottom",
        grid: {
          color: "#353F69",
        },
        ticks: {
          stepSize: 0.005,
          beginAtZero: true,
          callback: (value, index, values) => {
            return value.toFixed(3);
          },
          color: "#0d0f19",
        },
      },
      y: {
        type: "linear",
```

```

    min: -0.01,
    max: 0.03,
    position: "left",
    grid: {
      color: "#353F69",
    },
    ticks: {
      stepSize: 0.001,
      beginAtZero: true,
      callback: (value, index, values) => {
        return value.toFixed(3);
      },
      color: "#0d0f19",
    },
  },
},
elements: {
  point: {
    radius: 0,
  },
},
plugins: {
  legend: {
    display: true,
    labels: {
      color: "#ffffff",
    },
  }
},
},
};

```

```

$: if ($signs && isActive) {
  const newWidth = $signs.length * 5;
  chart.canvas.parentNode.style.width = `${
    newWidth < 1000 ? 1000 : newWidth
  }px`;
}

```

```
    box.scrollLeft = 0;

    chart.options.scales.x.min = $signs[0].x;
    chart.data.datasets[0].data = $signs;
    chart.update();
  }

  onMount(() => {
    chart = new Chart(canvas, config);
    isActive = true;
  });
</script>

<div class="root">
  <div class="box" bind:this={box}>
    <div class="container">
      <canvas bind:this={canvas}></canvas>
    </div>
  </div>
</div>

<style scoped lang="scss">
  div.root {
    width: 100%;
    height: 100%;

    display: grid;
    place-items: center;

    & .box {
      width: 100%;
      height: 100%;

      padding-bottom: 10px;

      scroll-behavior: smooth;
    }
  }
</style>
```



```
        overflow-x: scroll;

        & .container {
            width: 100%;
            height: 100%;
        }
    }
}
</style>

<script>
    import { onMount } from "svelte";

    export let logoutInvoked;
    export let itemInvoked;
    let selectedValue;

    let items = [
    { icon: "./assets/icon/history.svg", text: "History", value: "history" },
    { icon: "./assets/icon/ecg.svg", text: "Realtime", value: "realtime" },
    { icon: "./assets/icon/off.svg", text: "Logout", value: "logout" },
    ];

    function onItemClick(e, i) {
        // anulate default behavior
        e?.preventDefault();

        if (i === "logout") {
            logoutInvoked();
            return;
        }

        selectedValue = i;
        itemInvoked(i);
    }

    onMount(() => {
```

```

        onItemClick(null, "history");
    });
</script>

<div class="nav-bar">
  <a class="brand" href="/">
    
    <span>EBLU</span>
  </a>
  <nav class="nav">
    <ul>
      {#each items as { icon, text, value }}
        <li>
          <a
            class="nav-item"
            class:selected={selectedValue === value}
            on:click={(event) => onItemClick(event, value)}
            href="/"
          >
            <img src={icon} alt="" />
            <span>{text}</span>
          </a>
        </li>
      {/each}
    </ul>
  </nav>
</div>

<style scoped lang="scss">
  div.nav-bar {
    padding: 1.25rem 1.5rem;
    height: 4.5rem;

    display: flex;
    flex-direction: row;
    justify-content: space-between;
  }

```

```
a.brand {
  display: flex;
  flex-direction: row;
  align-items: center;
  gap: 0.75rem;

  text-decoration: none;

  & img {
    width: 2rem;
    height: 2rem;
  }

  & span {
    font-size: 1.5rem;
    font-weight: 600;

    color: var(--text-color);
  }
}
```

```
nav.nav {
  display: flex;
  flex-direction: row;
  align-items: center;

  & ul {
    display: flex;
    flex-direction: row;
    align-items: center;
    gap: 1.75rem;

    & li {
      list-style: none;
    }
  }
}
```

```
}
```

```
a.nav-item {
```

```
  display: flex;
```

```
  flex-direction: row;
```

```
  justify-content: center;
```

```
  align-items: center;
```

```
  gap: 0.5rem;
```

```
  text-decoration-color: blue;
```

```
  text-decoration: none;
```

```
  text-align: center;
```

```
  img {
```

```
    width: 100%;
```

```
    height: 100%;
```

```
  }
```

```
  & span {
```

```
    font-size: 1.1rem;
```

```
    color: var(--text-color);
```

```
    display: none;
```

```
  }
```

```
  &:hover {
```

```
    font-weight: 500;
```

```
  }
```

```
}
```

```
a.nav-item.selected {
```

```
  & img {
```

```
    filter: invert(0.5) sepia(1) saturate(3) hue-rotate(100deg);
```

```
  }
```

```
  & span {
```

```

        color: #5fffa0;
    }
}

@media only screen and (min-width: 600px) {
    a.nav-item {
        & span {
            display: block !important;
        }
    }
}
</style>

<script>
    import Chart from "chart.js/auto";

    import { onMount } from "svelte";

    import { signs } from "../stores";

    let isActive = false;
    let canvas;
    let chart;
    let box;

    // config...
    const config = {
        type: "line",
        data: {
            datasets: [
                {
                    label: "Cuadrícula X = 5 mS, Cuadrícula Y = 1 mV",
                    borderColor: "#59ed68",
                    borderWidth: 1.5,
                    radius: 0,
                },
            ],
        },
    ],

```

```
},
options: {
  responsive: true,
  maintainAspectRatio: false,
  showTooltips: false,
  animation: false,
  scales: {
    x: {
      type: "linear",
      min: 0,
      max: 1,
      position: "bottom",
      grid: {
        color: "#353F69",
      },
      ticks: {
        stepSize: 0.005,
        beginAtZero: true,
        callback: (value, index, values) => {
          return value.toFixed(3);
        },
        color: "#0d0f19",
      },
    },
  },
  y: {
    type: "linear",
    min: -0.01,
    max: 0.03,
    position: "left",
    grid: {
      color: "#353F69",
    },
    ticks: {
      stepSize: 0.001,
      beginAtZero: true,
      callback: (value, index, values) => {
        return value.toFixed(3);
      }
    }
  }
}
```

```

        },
        color: "#0d0f19",
    },
},
elements: {
    point: {
        radius: 0,
    },
},
plugins: {
    legend: {
        display: true,
        labels: {
            color: "#ffffff",
        },
    }
},
},
};

```

```

$: if ($signs && isActive) {
    chart.data.datasets[0].data = $signs;
    chart.update();
}

```

```

onMount(() => {
    chart = new Chart(canvas, config);
    isActive = true;
});

```

</script>

<div class="root">

<div class="box" bind:this={box}>

<div class="container">

<canvas bind:this={canvas}></canvas>

</div>

```
</div>
```

```
</div>
```

```
<style scoped lang="scss">
```

```
div.root {
```

```
width: 100%;
```

```
height: 100%;
```

```
display: grid;
```

```
place-items: center;
```

```
& .box {
```

```
width: 100%;
```

```
height: 100%;
```

```
padding-bottom: 10px;
```

```
scroll-behavior: smooth;
```

```
overflow-x: scroll;
```

```
& .container {
```

```
width: 100%;
```

```
height: 100%;
```

```
}
```

```
}
```

```
}
```

```
</style>
```



## **ANEXO D: ENTREVISTA AL MÉDICO**

### **Preguntas para validación del prototipo**

#### **DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO DE ELECTROCARDIÓGRAFO MÓVIL UTILIZANDO REDES MÓVILES PARA MONITORIZACIÓN DE LA FRECUENCIA CARDIACA EN TIEMPO REAL**

**Objetivo:** Validar resultados obtenidos del prototipo, para una mejora futura del mismo

**1. ¿Cuál es su punto de vista que presenta el prototipo?**

La fabricación de este prototipo es una idea innovadora, ya que en el campo de la cardiología, este tipo de dispositivos suelen ser muy costosos e inaccesibles para muchos pacientes. Esto puede hacer que estos dispositivos se vuelvan más comunes y accesibles, y puedan ser utilizados incluso en centros de salud alejados, donde es difícil encontrar especialistas en cardiología.

**2. ¿Cuáles serían las ventajas que presenta el prototipo?**

La principal ventaja de este dispositivo es su bajo costo, lo que lo hace accesible para adquirirlo. Esto permite que cada paciente pueda realizar un examen y ser monitoreado por su médico tratante sin necesidad de salir de casa. Como resultado, los exámenes pueden realizarse con mayor regularidad, lo que facilita la detección temprana de alteraciones cardíacas.



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**CERTIFICADO DE CUMPLIMIENTO DE LA GUÍA PARA**  
**NORMALIZACIÓN DE TRABAJOS DE FIN DE GRADO**

**Fecha de entrega:** 15/05/2024

<b>INFORMACIÓN DEL AUTOR</b>
<b>Nombres – Apellidos:</b> Guillermo Alejandro Pinguil Duchí
<b>INFORMACIÓN INSTITUCIONAL</b>
<b>Facultad:</b> Facultad de Informática y Electrónica
<b>Carrera:</b> Telecomunicaciones
<b>Título a optar:</b> Ingeniero en Telecomunicaciones
 Jefferson Ribadeneira Ramirez <b>Director del Trabajo de Titulación</b>
 Mayra Alejandra Pacheco Cunduri <b>Asesor del Trabajo de Titulación</b>