

Diseño de una plataforma basada en las tecnologías IoT para el registro y monitoreo remoto de temperatura, ritmo cardiaco y oxígeno en sangre, de un paciente en cuidados fuera de instalaciones médicas.

Gabriel Felipe Vega Ochoa y Luis Miguel Amaya García

Trabajo de Grado para Optar al Título de Ingeniero de Sistemas

Director

Jose Geralbert Rubiano

Especialista en Redes de Computadoras

Codirector

Laura Isabel Valencia Angel

Magíster en Epidemiología

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería de Sistemas

Bucaramanga

2023

### **Dedicatoria**

Dedicado a toda mi familia, porque cada uno de ellos es la razón de lo que soy, y serán la razón de cada logro que pueda alcanzar.

#### **Luis Miguel Amaya García**

Quiero dedicar este trabajo a mis padres por apoyarme y guiarme durante el proceso de cursar mi carrera, ser el soporte en los momentos difíciles e impulsarme a siempre realizar las cosas de la mejor manera, y siempre mostrarme su orgullo por los logros que he obtenido.

A mi abuela Ana Socorro en agradecimiento por haber sido esa persona que durante toda mi vida me ha apoyado y me ha formado, permitiéndome ser la persona que soy hoy en día, siendo uno de los mayores pilares que he tenido.

A Dios cómo guía y soporte de mis procesos.

#### **Gabriel Felipe Vega Ochoa**

### **Agradecimientos**

Agradezco a cada uno de los integrantes de mi familia, porque con ellos siempre tengo paz y felicidad, y esto siempre me da motivación para lograr cada uno de mis objetivos.

Agradezco a mi compañero Gabriel Vega, por permitirme hacer parte de este proyecto innovador, y que por su gran carácter, humildad y sabiduría lo considero un gran ejemplo de colega.

#### **Luis Miguel Amaya García**

Agradezco a mis padres por su apoyo, su consejo y su ayuda durante toda mi vida universitaria, siendo ellos gran parte del mérito de este logro, sin ellos hubiese sido imposible, a mis familiares, tíos, primos, abuelas por su apoyo en la consecución de este logro, a mis amigos que han sido un apoyo muy importante. A mi compañero Luis Miguel Amaya por su trabajo y dedicación a este proyecto, asumiendo muchas de las responsabilidades que nos han permitido culminar satisfactoriamente el mismo.

A la Universidad Industrial de Santander, profesores y compañeros que me han dado las herramientas para llevar a cabo este proyecto, el conocimiento y la formación necesaria.

#### **Gabriel Felipe Vega Ochoa**

Finalmente queremos agradecer al profesor Jose Geralbert Rubiano por su guía, comprensión y apoyo en el transcurso del proyecto, así mismo a la Doctora Laura Isabel Valencia Angel por su apoyo y buena disposición para ayudarnos en la resolución de este proyecto.

**Tabla de Contenido**

	<b>Pág.</b>
Introducción	14
1. Planteamiento y Justificación Del Problema	15
2. Objetivos	19
2.1 Objetivo General	1
2.2 Objetivos Específicos	19
3. Marco Referencial	19
3.1 Marco Teórico	19
3.1.1 Electrocardiograma	19
3.1.2 Arritmia cardíaca	20
3.1.2 Oximetría de pulso por transmisión o convencional.	22
3.1.3 Oximetría de pulso por reflectancia	22
3.1.3 Interpretación de resultados de pruebas de pulsioximetría	23
3.1.3 Temperatura como factor de riesgo cardiovascular	24
3.1.4 Microcontrolador	26
3.1.5 Internet de las cosas (IOT)	26
3.1.6 Arquitectura de software	26
3.1.7 Bluetooth de baja energía	27
3.1.8 Sensor de temperatura de tipo infrarrojo	27
3.1.9 Computación en la Nube	28
3.1.10 Serverless Computing	29
3.1.11 Infraestructura Como Servicio	30
3.1.12 KiCad	30
3.1.13 Peticiones HTTP	31
3.1.14 Respuestas HTTP	31
3.1.15 Protocolo HTTPS	31
3.1.16 Certificado SSL	31
3.1.17 API REST	32
3.1.18 Librería de programación	32
3.1.19 React	32
3.2 Estado del Arte	33
3.2.1 Huawei Health	33
3.2.2 Diseño y construcción de un sistema basado en IoT para supervisión remota de la frecuencia cardíaca y el tono simpático a partir de la señal fotopletoislográfica (PPG).	33
3.2.3 Monitoreo del ritmo cardíaco a través de dispositivos móviles	33

4. Marco Metodológico	34
4.1 Obtención de requerimientos	34
4.2 Capacitación	34
4.1 Análisis y selección de arquitectura	35
4.2 Diseño	35
4.3 Desarrollo del prototipo	35
4.4 Evaluación y pruebas de prototipo	35
4.5 Entrega de prototipo final	36
5. Desarrollo del Proyecto	36
5.1 Obtención de requerimientos	36
5.2 Capacitación	37
5.3 Análisis y selección de arquitectura	38
5.3.1 Análisis de alcance, viabilidad y riesgos	38
5.3.2 Identificación de componentes y factores	39
5.3.3 Evaluación de condiciones y límites para desarrollo de los componentes	40
5.3.4 Planteamiento de arquitectura	40
5.3.4.1 Amazon Web Services (AWS).	42
5.3.4.2 Google Cloud Platform.	43
5.4 Diseño	47
5.4.1 Realización de estimaciones	47
5.4.2 Identificación y asignación de recursos	49
5.4.3 Establecimiento de ruta de trabajo	50
5.4.4 Análisis de datos y optimización de modo de trabajo	51
5.5 Desarrollo del prototipo	53
5.5.1 Establecimiento del entorno de trabajo	53
5.5.2 Asignación de tareas	54
5.5.2.1 Búsqueda y selección de los componentes de hardware.	54
5.5.2.2 Adquisición y prueba de los componentes hardware.	54
5.5.2.3 Implementación y prueba del entorno base en la nube.	54
5.5.2.4 Selección y pruebas iniciales de la tecnología front-end.	55
5.5.2.5 Montaje inicial de componentes hardware en placa de pruebas.	55
5.5.2.6 Desarrollo y pruebas de adquisición de datos.	55
5.5.2.7 Diseño y desarrollo front-end.	55
5.5.2.8 Exploración e integración de servicios en la nube con front-end.	55
5.5.2.9 Definición y desarrollo de modo de conexión entre el dispositivo hardware y la aplicación móvil para transmisión de datos.	56
5.5.2.10 Integración completa entre dispositivo hardware, aplicación móvil, servicios en la nube	

y web.	56
5.5.2.11 Pruebas de funcionamiento a nivel de desarrollo y corrección de errores.	56
5.5.2.12 Mejoras, optimización y refactor general del código.	56
5.5.3 Ejecución de tareas	57
5.5.3.1 Búsqueda y selección de los componentes de hardware.	57
5.5.3.2 Adquisición y prueba de los componentes hardware.	58
5.5.2.3 Implementación y prueba del entorno base en la nube.	60
5.5.2.4 Selección y pruebas iniciales de la tecnología front-end.	62
5.5.2.5 Montaje inicial de componentes hardware en placa de pruebas.	63
5.5.2.6 Desarrollo y pruebas de adquisición de datos. Se realizó el desarrollo para la placa de desarrollo ESP32 Devkit en el cuál permitiera adquirir los datos de las variables definidas	66
5.5.2.7 Diseño y desarrollo front-end. Ya que se cuenta con dos proyectos frontend que serán parte de la plataforma, se dividió el desarrollo en estos, la aplicación móvil y la plataforma web.	66
5.5.2.7.1 Aplicación móvil. Se definieron algunas pantallas iniciales, para la aplicación móvil, que fueron desarrolladas en React Native, como previamente se definió.	66
5.5.2.7.2 Plataforma web. Para la plataforma web se definió que la pantalla principal fuese el listado de las aplicaciones y un pantalla secundaria accesible por medio del menú superior la cuál muestra el listado de usuarios editable.	67
5.5.2.8 Exploración e integración de servicios en la nube con front-end.	69
5.5.2.9 Definición y desarrollo de modo de conexión entre el dispositivo hardware y la aplicación móvil para transmisión de datos.	70
5.5.2.10 Integración completa entre dispositivo hardware, aplicación móvil, servicios en la nube y web.	71
5.6 Evaluación y pruebas del prototipo	73
5.6.1 Realización de pruebas, toma de datos	73
5.6.1.1 Presentación de funcionalidades del prototipo.	74
5.6.1.2 Interacción del público con el prototipo.	75
5.6.1.3 Toma de datos de voluntarios.	78
5.6.1.4 Pruebas mediciones registradas en web.	81
5.6.1.5 Pruebas de errores en la página web.	84
5.6.1.6 Pruebas de errores en la aplicación móvil. E	86
5.6.1.7 Comparación entre dispositivos.	88
5.6.2 Recepción de críticas y sugerencias	90
5.6.2.1 Críticas del público general (estudiantes).	90
5.6.2.2 Críticas del experto (Dra. Laura Valencia).	92
5.6.3 Interpretación de información y éxito del prototipo	92
6. Conclusiones	93
7. Recomendaciones	95

Referencias Bibliográficas

98

Apéndices

107

**Lista de Tablas**

	<b>Pág.</b>
Tabla 1. Precios de los proveedores de servicios en la nube Amazon Web Service (AWS) y Google Cloud Platform (GCP)	46
Tabla 2. Componentes previstos para la construcción del prototipo con su precio actual.	49
Tabla 3. Comparativa de medición de pulsos por minuto [BPM] tomados por el pulsioxímetro de uso médico y por el sensor de pulsioximetría del prototipo.	88
Tabla 4. Comparativa de medición de oxígeno en sangre [spO2] tomados por pulsioxímetro de uso médico y por el sensor de pulsioximetría del prototipo.	89
Tabla 5. Comparativa de medición de grados celsius [°C] tomados por termómetro de uso médico y por el sensor del prototipo	89

**Lista de Figuras**

Figura 1. Diagrama del funcionamiento eléctrico del corazón	20
Figura 2. Ilustración equivalencia comportamientos del corazón con su representación en un electrocardiograma	22
Figura 3. Ilustración dispositivo de pulsioximetría	24
Figura 4. Ilustración rangos de temperatura para una persona	25
Figura 5. Esquema de funcionamiento básico de un sensor infrarrojo	28
Figura 6. Esquema de referencia para concepto de computación en la nube	29
Figura 7. Diagrama distribución de servicios ofrecidos en la nube	30
Figura 8. Esquema de metodología de trabajo	34
Figura 9. Diagrama de casos de uso	39
Figura 10. Comparación entre arquitectura tradicional y sin servidor	41
Figura 11. Tabla de precios implementación de componentes en la nube AWS	42
Figura 12. Tabla de precios implementación de componentes en la nube GCP	44
Figura 13. Diagrama de arquitectura, solución serverless basada en los servicios de AWS	47
Figura 14. Diseño placa PCB para montaje de circuito	50
Figura 15. Fotografías de componentes utilizados para realización de dispositivo hardware	58
Figura 16. Fotografía placa PCB fabricada	58
Figura 17. Panel de manejo principal del servicio AWS Amplify	60
Figura 18. Bloque de código correspondiente a una función lambda implementada en node.js	61
Figura 19. Captura de pantalla panel de administración con detalle de API Gateway de AWS	62
Figura 20. Montaje en protoboard de prototipo dispositivo hardware	63
Figura 21. Modelo 3D final del dispositivo hardware sin ensamblar	64
Figura 22. Montaje final dispositivo hardware terminado y ajustado en la superficie de la mano	65
Figura 23. Consola de monitor serial con impresión de las variables capturadas por el dispositivo	66
Figura 24. Pantallas diseñadas para funcionamiento de la aplicación móvil	66
Figura 25. Pantalla principal de plataforma web que muestra listado de mediciones	67
Figura 26. Pantalla de usuarios de plataforma web	68
Figura 27. Comandos utilizados para la implementación de AWS Amplify en React	70
Figura 28. Bloques de código para la implementación de AWS Amplify en React	70
Figura 29. Pantalla de aplicación móvil con historial de mediciones	72
Figura 30. Captura de pantalla panel de administración de DynamoDB mostrando la tabla de registros de mediciones	72

Figura 31. Fotografía de miembros del proyecto en presentación ante estudiantes de medicina	74
Figura 32. Fotografía de miembros del proyecto junto con estudiantes de medicina	75
Figura 33. Fotografía de miembros del proyecto en presentación ante estudiantes de medicina	76
Figura 34. Toma de pulso a modelo de persona en simulador de parto	77
Figura 35. Dispositivos para simulación en sala de parto	77
Figura 36. Fotografía voluntario en toma de mediciones	78
Figura 37. Fotografía prototipo y pulsioxímetro comercial realizando mediciones en paralelo	79
Figura 38. Fotografía co-tutora y miembro del proyecto, en toma de datos con prototipo y pulsioxímetro comercial	80
Figura 39. Fotografía de toma de temperatura por parte del sensor del prototipo	81
Figura 40. Medición en la plataforma web heart monitor, tomada en ejecución de las pruebas tomada debido a las alertas	82
Figura 41. Lista de mediciones registradas en la plataforma web heart monitor	82
Figura 42. Visualización del gráfico cardíaco de una medición de en la plataforma web	83
Figura 43. Carga de registros cardíacos con un error	84
Figura 44. Carga de listado de usuarios con un error	84
Figura 45. Guardado de modificación de usuarios con un error	85
Figura 46. Error por contraseña incorrecta login	85
Figura 47. Error por problemas de red en el login	86
Figura 48. Error por problemas de red en el login en la aplicación móvil	87
Figura 49. Error generado por ingreso erróneo de las credenciales en el login	87

**Lista de Apéndices**

Apéndice A. Documento plan de pruebas ritmo cardiaco y saturación	107
Apéndice B. Documento plan de pruebas temperatura y observaciones	108
Apéndice C. Documento plan de pruebas observaciones	109

## Glosario

**Alerta:** notificación o aviso de un evento o condición importante.

**Análisis de datos:** proceso de examinar y extraer información significativa de conjuntos de datos.

**App:** aplicación móvil.

**Cardiovascular:** relativo al sistema cardiovascular o al corazón.

**Conectividad:** capacidad de establecer y mantener una conexión a Internet.

**Interfaz de usuario:** espacio o medio a través del cual el usuario interactúa con la aplicación o el dispositivo IoT.

**Monitoreo:** supervisión o seguimiento constante.

**Notificación:** mensaje o aviso que se muestra al usuario.

**Plataforma en la nube:** infraestructura en línea que permite el almacenamiento y procesamiento de datos en servidores remotos.

**Ritmo cardíaco:** frecuencia de los latidos del corazón.

**Sensor:** dispositivo que detecta y registra datos.

**Usuario:** persona que utiliza la aplicación o el dispositivo IoT para el monitoreo cardiovascular.

## Resumen

**Título:** Diseño de una plataforma basada en las tecnologías IoT para el registro y monitoreo remoto de temperatura, ritmo cardiaco y oxígeno en sangre, de un paciente en cuidados fuera de instalaciones médicas.<sup>1\*</sup>

**Autor:** Gabriel Felipe Vega Ochoa y Luis Miguel Amaya Garcia<sup>2\*3\*</sup>

**Palabras Clave:** Plataforma IoT, Salud, Computación en la nube, Monitoreo cardiovascular, Microcontroladores

### Descripción:

Pacientes con enfermedades cardiovasculares requieren un estricto control sobre el estado de sus signos vitales, principalmente el comportamiento cardiovascular, temperatura y oxígeno en la sangre. Aún no se pueden calcular con certeza la cantidad de muertes que se habrían podido evitar con automonitoreo cardiovascular o monitoreo supervisado por un subresponsable del paciente, sin embargo, 628.630 personas en Colombia perdieron la vida por problemas cardiovasculares entre 1998 y 2011 que corresponden al 23.5% del total de muertes en Colombia en este periodo (Observatorio Nacional de Salud & Instituto Nacional de Salud., 2013).

Durante la pandemia provocada por Covid-19, los problemas cardiovasculares se posicionaron como una comorbilidad supremamente susceptible y fatal para aquellas personas que la sufren y que son propensas a contraer dicho virus, y se calcula que en Colombia alrededor de 1.924 (Díaz, J. E., 2020) personas identificadas con comorbilidades cardiovasculares, fallecieron durante el desarrollo de la pandemia, esto posiblemente causado por escasez de camas UCI, restricciones de movilidad, acceso limitado a hospitales, entre otros.

Los instrumentos de medición existentes resultan de un alto costo y muchos de los dispositivos del mercado que permiten la monitorización de datos vitales suelen ser de propósito general, no orientado a la salud, o limitados en la accesibilidad de los datos tomados.

¿Existe alguna plataforma que permita monitorear y registrar variables en pacientes con comorbilidades que se encuentren fuera de un centro de salud?, Partiendo de esta pregunta, en este proyecto se diseñó una plataforma IoT para el monitoreo y registro de variables en pacientes con problemas cardiovasculares bajo cuidados domiciliarios, permitiendo mediante un dispositivo electrónico medir variables como temperatura, ritmo cardiaco y oxígeno en sangre, permitiendo llevar un registro de los pacientes así como, la posibilidad de fijar niveles de alarma sobre estas variables, lo cual permite una respuesta oportuna ante una emergencia.

---

<sup>1\*</sup> Trabajo de Grado

<sup>2\*\*</sup> Facultad de Ingenierías Físico-mecánicas. Escuela de Ingeniería de Sistemas e Informática. Director: Jose Geralbert Rubiano, Especialista en Redes de Computadoras Codirector: Laura Isabel Valencia Angel, Magíster en Epidemiología

### Abstract

**Title:** Design of a platform based on IoT technologies for remote recording and monitoring of temperature, heart rate and blood oxygen level of a patient in out-of-facility care.<sup>4\*</sup>

**Author(s):** Gabriel Felipe Vega Ochoa y Luis Miguel Amaya Garcia<sup>5</sup>

**Key Words:** IoT platform, Health, Cloud computing, Cardiovascular monitoring, Microcontrollers

### Description:

Patients with cardiovascular diseases require strict control over the status of their vital signs, mainly cardiovascular behavior, temperature and blood oxygen. It is not yet possible to calculate with certainty the number of deaths that could have been avoided with cardiovascular self-monitoring or monitoring supervised by a person responsible for the patient, however, 628,630 people in Colombia lost their lives due to cardiovascular problems between 1998 and 2011, which corresponds to 23.5% of the total deaths in Colombia in this period (National Health Observatory & National Institute of Health., 2013).

During the pandemic caused by Covid-19, cardiovascular problems were positioned as a supremely susceptible and fatal comorbidity for those people who suffer from it and who are prone to contracting said virus, and It is estimated that in Colombia around 1,924 (Díaz, J. E., 2020) people identified with cardiovascular comorbidities died during the development of the pandemic, possibly caused by a shortage of ICU beds, mobility restrictions, limited access to hospitals, among others.

Existing measurement instruments resulting in high cost and many of the devices on the market that allow the monitoring of vital data are usually general purpose, not health-oriented, or limited in the accessibility of the data taken.

Is there a platform that allows monitoring and recording variables in patients with comorbidities who are outside a health center? Based on this question, in this project an IoT platform was developed for monitoring and recording variables in patients with cardiovascular problems. under home care, allowing through an electronic device to measure variables such as temperature, heart rate and blood oxygen, allowing to carry out a record of patients as well as the possibility of setting alarm levels on these variables, which allows a timely response in an emergency

---

<sup>4\*</sup> Degree Work

<sup>5</sup>Faculty of Physical-mechanical Engineering. School of Systems Engineering and Informatics. Director: Jose Geralbert Rubiano, Specialist in Computer Networks Co-director: Laura Isabel Valencia Angel, Master in Epidemiology

## Introducción

Las enfermedades cardiovasculares y/o respiratorias representan un factor de importancia dentro del contexto de salud colombiano, pues en 2019 se tuvieron 38.475 muertes relacionadas con enfermedades isquémicas del corazón y 15.543 a enfermedades cerebrovasculares de un total de 242.609, según las cifras presentadas por el Dane (Bayer, 2020). Adicional a esto se ha sumado el contexto de pandemia en el que se buscaba tener menor cantidad de pacientes en los hospitales, además de surgir consecuencias respiratorias a raíz del covid-19 (Díaz-Castrillón, F. J., & Toro-Montoya, A. I., 2020).

En este contexto surge la necesidad de monitoreo hacia los pacientes sin estar directamente en un establecimiento dónde se presten servicios de salud, y con esto se han buscado soluciones que suplan esta necesidad, sin embargo, no se ha logrado establecer un sistema completo que permita la monitorización.

El internet de las cosas, cómo escenario dónde se extiende la capacidad de cómputo y la conectividad a objetos de uso diario (Rose, K., Eldridge, S., & Chapin, L., 2015), se postula cómo una herramienta importante para soluciones de salud debido a la capacidad de adquisición de datos en tiempo real a bajo costo y con independencia energética (Rodríguez-Gómez, R., 2019).

Con esto entonces se propone el diseño de una plataforma basada en las tecnologías IoT que permita el registro y monitoreo remoto de variables vitales, las cuales son temperatura, ritmo cardiaco y oxígeno en sangre.

Es posible encontrarse con una gran diversidad de aplicativos y dispositivos asociados a muchas de estas que cumplen una función similar, pero con un propósito meramente informativo

más no de alerta y de prevención como sí lo propone este proyecto; de la misma manera, los costos para la accesibilidad a estas plataformas que cuentan con un dispositivo asociado, son elevados y ofrecen funciones básicas como la medición de ritmo cardiaco, y en el caso de medición de temperatura normalmente se encuentran sólo termómetros de único propósito y para la toma de oxígeno en sangre se encuentra de manera independiente los oxímetros.

Esta plataforma sí tiene como propósito a través de la medición de estos tres factores, analizar e interpretar los datos para informar mediante la plataforma el estado del paciente, y alertar en caso de riesgo, teniendo como ventaja un bajo costo de fabricación y alto grado de efectividad para aplicar a pacientes con condiciones y enfermedades de tipo cardiovascular.

La implementación de este proyecto, aplicándose adecuadamente a pacientes con problemas cardiovasculares y otras condiciones que se manifiesten como comorbilidades, ayudará a reducir la tasa de mortalidad o incidencias causadas por aspectos como carencia de camas UCI, negligencia médica, falta de control médico, entre otros.

## **1. Planteamiento y Justificación Del Problema**

A día de hoy, las dificultades y carencias que vive la salud en Colombia, se debe en parte a que se presenta un retraso en la modernización y adopción de nuevas prácticas que facilitan ya en otras regiones el control de los pacientes tanto para el autocuidado como para el seguimiento por parte de los profesionales de la salud.

Especialmente se puede evidenciar lo anteriormente mencionado en pacientes con enfermedades cardiovasculares y/o respiratorias diagnosticadas, puesto que son estos quienes

requieren un mayor control sobre el estado de sus signos vitales, cómo el ritmo cardíaco, el comportamiento del sistema cardiovascular y los porcentajes de oxígeno en la sangre.

Aún no se pueden calcular con certeza la cantidad de muertes que se habrían podido evitar con tan sólo un auto-monitoreo cardiovascular o ya sea también supervisado por un sub-responsable del paciente, sin embargo, se calcula que 628.630 personas en Colombia han perdido la vida por problemas cardiovasculares entre 1998 y 2011 que corresponden al 23.5% del total de muertes en Colombia en este periodo (Observatorio Nacional de Salud & Instituto Nacional de Salud., 2013). Si bien se conoce que muchos de aquellos problemas son descendientes de un daño secundario o generan más comorbilidades en el cuerpo humano, al contar con un monitoreo en tiempo real del comportamiento de sus signos vitales, especialmente su corazón, podría evitar una cantidad innumerable de calamidades por descuidos, falta de control o causales de falta de tiempo para proceder a una intervención quirúrgica inmediata.

Adicionalmente, la no detección temprana de enfermedades cardiovasculares, o la aparición de falsos negativos en exámenes especializados representa un riesgo latente para personas con estas condiciones, y las cuáles podrían recibir un diagnóstico más adecuado si se realiza un monitoreo más extenso que permita verificar su estado bajo diferentes condiciones durante el desarrollo de un periodo de tiempo.

Es de resaltar que, durante la pandemia provocada por Covid-19, los problemas cardiovasculares se posicionaron cómo una comorbilidad supremamente susceptible y fatal para aquellas personas que la sufren y que son propensas a contraer dicho virus, el el informe del centro de control y prevención de enfermedades de China con 72.314 casos confirmados de covid el 10.5% de pacientes que presentaban enfermedad cardiovascular fallecieron (Triana, F. J.

F. ,2020), siendo así el segundo factor de riesgo más relevante sólo después de las personas mayores de 80 años.

Ubicándonos en el contexto nacional, se calcula que en Colombia alrededor de 1.924 (Díaz, J. E., 2020). personas identificadas con comorbilidades cardiovasculares, fallecieron durante el desarrollo de la pandemia, esto desglosa una lista de razones posibles para haber terminado en dicha situación, como por ejemplo, escasez de camas UCI, restricciones de movilidad y acceso limitado a hospitales, estancamiento de citas de control y seguimiento a pacientes, prohibición de prácticas deportivas al aire libre, difícil acceso a dispositivos de monitoreo, desconocimiento de herramientas y prácticas de autocuidado. En adición a esto, surgió una demanda de dispositivos de oximetría (Herrera, K. A., 2020) por la necesidad de monitorear los niveles de oxígeno en pacientes diagnosticados covid, así cómo para realizar o descartar diagnósticos relacionados.

En torno a las soluciones existentes para estas problemáticas se evidencia que los instrumentos de medición existentes resultan en un alto costo para los centros de salud y/o los pacientes que los necesiten, adicionalmente muchos de los dispositivos existentes en el mercado que permiten la monitorización de ciertos datos vitales suelen ser de propósito general, no orientado a la salud, o limitados en cuanto a la accesibilidad de los datos tomados. Y teniendo en cuenta que los instrumentos de medición cardiovasculares representan el segundo lugar de cuota de mercado dentro de los dispositivos de tecnología médica (EvaluateMedTech., 2018). Colombia ocupa el tercer lugar en importaciones de dispositivos médicos en Latinoamérica(Urbina, I. & El Hospital. ,2016) se puede identificar un déficit en la disponibilidad de sistemas de medición especializados para la salud en el país, así cómo una creciente demanda global que podría agravar esta situación y las anteriormente mencionadas.

De lo expuesto anteriormente surgen unas interrogantes. ¿Existe alguna plataforma que permita monitorear y registrar variables en pacientes con comorbilidades que se encuentren fuera de un centro de salud?, ¿Podemos utilizar plataformas IoT para realizar monitoreo remoto de pacientes? De estas inquietudes nace el objetivo principal de este trabajo en donde se diseñará una plataforma IoT para el monitoreo y registro de variables en pacientes con problemas cardiovasculares bajo cuidados domiciliarios, la cual permitirá mediante un dispositivo electrónico medir variables como temperatura, ritmo cardíaco y porcentaje de oxígeno en sangre, permitiendo llevar un registro de los pacientes así como, la posibilidad de fijar niveles de alarma sobre estas variables, lo cual que permita una respuesta oportuna ante una emergencia.

## 2. Objetivos

### 2.1 Objetivo General

Diseñar una plataforma basada en las tecnologías IoT para el registro y monitoreo remoto de temperatura, ritmo cardiaco y oxígeno en sangre, de un paciente en cuidados fuera de instalaciones médicas.

### 2.2 Objetivos Específicos

Identificar las características de las variables, hardware y requerimientos funcionales y no funcionales involucrados en el diseño de la plataforma.

Seleccionar la arquitectura de hardware y software para el desarrollo del diseño.

Plantear una arquitectura de software para una plataforma que permita la integración de dispositivos de medición y monitoreo con un entorno de monitoreo remoto.

Diseñar la plataforma conforme a la arquitectura software y tecnologías identificadas.

Evaluar mediante la elaboración de un prototipo la interacción del hardware y software diseñados.

### 3. Marco Referencial

#### 3.1 Marco Teórico

##### 3.1.1 Electrocardiograma

El electrocardiograma (ECG) es la señal que permite describir el comportamiento eléctrico del corazón. Esta señal está compuesta por una adición de ondas que conforman el complejo cardíaco: onda P, onda Q, onda R, onda S y onda T, tal como se pudo observar en la figura.

Las diferentes ondas que componen el complejo cardíaco reflejan la activación eléctrica del miocardio auricular y ventricular, así como la repolarización ventricular. La actividad eléctrica del nodo sino auricular (sinusal), nodo auriculoventricular (AV), el Haz de his y sus ramas, no es registrada en el electrocardiograma, pero mediante el análisis de la secuencia de la propagación del impulso se hace factible la deducción del comportamiento electrofisiológico de dichas estructuras (Koeppen B, 2009). La figura 1 permite observar la relación de cada onda del complejo cardíaco con los componentes del sistema eléctrico del corazón.

#### Figura 1

Diagrama del funcionamiento eléctrico del corazón



Nota. Adaptado de Electrocardiograma desde una visión digital (p. 34). por J. Gonzáles, 2016, Ediciones USTA.

### 3.1.2 Arritmia cardíaca

Una arritmia cardíaca es un latido irregular del corazón. Los problemas del ritmo cardíaco (arritmias cardíacas) ocurren cuando los impulsos eléctricos que coordinan los latidos del corazón no funcionan adecuadamente. La señalización defectuosa hace que el corazón lata demasiado rápido (taquicardia), demasiado lento (bradicardia) o de forma irregular.

Las arritmias cardíacas pueden hacer que un paciente tenga una sensación de aleteo o de corazón acelerado y pueden ser inofensivas. Sin embargo, algunas arritmias cardíacas pueden provocar signos y síntomas molestos y, que, a veces, ponen en riesgo la vida; Sin embargo, a veces es normal que una persona tenga un ritmo cardíaco rápido o lento. Por ejemplo, la frecuencia cardíaca puede aumentar con el ejercicio o disminuir durante el sueño.

El tratamiento de las arritmias cardíacas puede incluir medicamentos, procedimientos con catéter, dispositivos implantados o cirugía para controlar o eliminar los latidos cardíacos rápidos, lentos o irregulares. Un estilo de vida saludable para el corazón puede ayudar a prevenir los daños cardíacos que pueden desencadenar ciertas arritmias cardíacas.

Los tipos de arritmia cardiaca son lo siguientes:

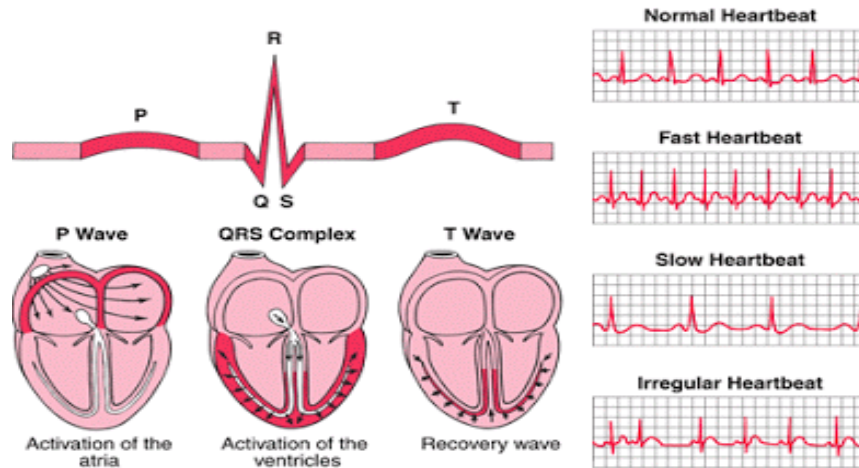
- *Taquicardia*: es un latido rápido del corazón. La frecuencia cardíaca en reposo es superior a 100 latidos por minuto.
- *Bradicardia*: es un latido lento del corazón. La frecuencia cardíaca en reposo es inferior a 60 latidos por minuto.

(Mayo Foundation for Medical Education and Research, 2022)

La figura 2 representa diferentes ejemplos de ritmos cardíacos (ver Figura 2).

**Figura 2**

Ilustración equivalencia comportamientos del corazón con su representación en un electrocardiograma.



Nota. Tomado de Arritmia cardíaca y acupuntura, un estudio y un caso personal, por M. Makarian.

### ***3.1.2 Oximetría de pulso por transmisión o convencional.***

Es un método simple, continuo, no invasivo, para vigilar de manera periférica el porcentaje de hemoglobina (Hb) saturada con oxígeno (O<sub>2</sub>), por el paso de longitudes de onda específicas a través de la sangre (SpO<sub>2</sub>). Se basa en la ley de Lambert-Beer-Bouguer. Ley de Beer-Lambert: El principio en el que se funda la determinación de la saturación de O<sub>2</sub>, con el oxímetro de pulso, es la ley de Beer. Todas las técnicas de oximetría se basan en análisis espectrofotométricos que miden las porciones de luz transmitida y absorbida por la hemoglobina, combinado con el principio de la pletismografía.

### ***3.1.3 Oximetría de pulso por reflectancia***

La RSpO<sub>2</sub> se sustenta en la reflexión de la sangre en los tejidos. Es un método no invasivo que proporciona un monitoreo continuo, exacto, confiable y económico de la saturación

de O<sub>2</sub>, con el empleo de la tecnología de reflectancia. Los diodos emisores de luz y el fotodetector permiten la absorción en la superficie de la piel.

El sensor del oxímetro de pulso por reflectancia emite luz hacia abajo a través del tejido y del lecho capilar y recoge la luz que se refleja hacia arriba, y calcula la saturación de O<sub>2</sub>.

### ***3.1.3 Interpretación de resultados de pruebas de pulsioximetría***

Los resultados que arroja el oxímetro de pulso, se dan en porcentaje de saturación de oxígeno en la sangre. Un nivel de saturación de oxígeno normal oscila entre el 95 % y el 100 %. Los niveles de saturación pueden ser un poco más bajos y se consideran aceptables si el paciente tiene una enfermedad de tipo pulmonar como neumonía. Los niveles también pueden ser más bajos si el usuario vive en un lugar con mayor elevación sobre el nivel del mar.

Si el oxímetro de pulso que se usa es para el hogar y el nivel de saturación de oxígeno leído es del 92 % o menos, es importante comunicarse con un profesional de la salud; pero si éste baja al 88 % o menos, hay que solicitar atención médica inmediata ya que es un nivel crítico.

Los resultados arrojados por la pulsioximetría pueden ser entre 2% y 4% por encima o por debajo del nivel real de oxígeno en sangre que posee el paciente. Para obtener un resultado más preciso en caso de ser necesario, otra prueba importante es un análisis de sangre llamado gasometría arterial que puede practicar un profesional de la salud.

### **Figura 3**

Ilustración dispositivo de pulsioximetría



Nota. Tomado de Oxímetro de pulso o pulsioxímetro CDPC01 - C.D C.D.Products S.A.

### ***3.1.3 Temperatura como factor de riesgo cardiovascular***

Un descenso de las temperaturas puede provocar vasoconstricción en las arterias coronarias y una interrupción de la llegada de sangre al miocardio, elevando el riesgo de padecer un infarto.

Algunas patologías cardiovasculares empeoran debido a un cambio climático drástico. De esta forma, la variación estacional, especialmente en épocas de frío, puede influir en la actividad corporal modificando el tono simpático, la viscosidad sanguínea, la tensión arterial y la frecuencia cardíaca, que tienden a aumentar, incrementando a su vez la demanda de oxígeno del organismo. Con el descenso de las temperaturas, aquellas personas que tienen antecedentes cardiovasculares están más expuestas a sufrir complicaciones, ya que las patologías cardiovasculares están principalmente asociadas al frío. Cuando el cuerpo pierde calor más rápido de lo que lo produce se puede presentar una hipotermia, lo que provoca una peligrosa

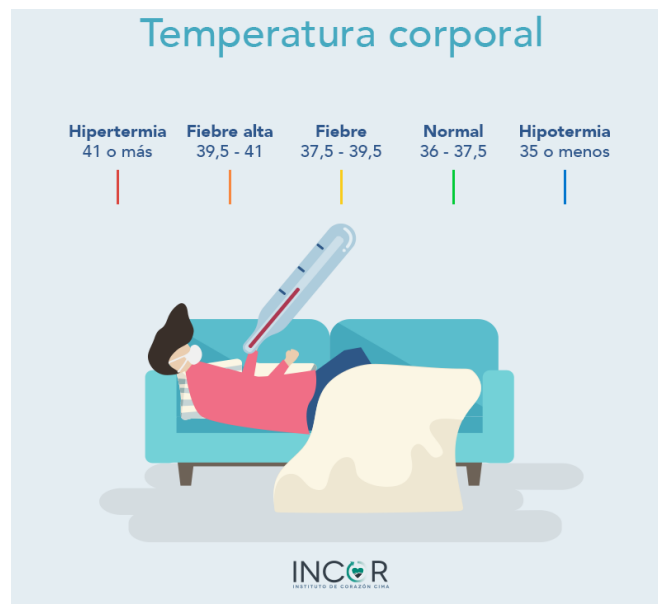
disminución de la temperatura corporal. La hipotermia se produce cuando la temperatura del cuerpo cae por debajo de los 35 °C.

Cuando desciende la temperatura corporal, el corazón, el sistema nervioso y otros órganos no pueden funcionar normalmente, y si no se trata prontamente, la hipotermia puede provocar una insuficiencia del corazón y del sistema respiratorio total y en algunos casos, ha provocado la muerte.

Por otra parte y referente a las altas temperaturas corporales, existe una relación entre los infartos de miocardio y algunas infecciones del sistema respiratorio, lo que para un incremento de temperatura de 2°C, correspondiente a una fiebre de 39°C, resulta suficientemente elevado como para ser tenido en cuenta ya que contribuye al síndrome coronario agudo. Dichos cambios en la temperatura corporal pueden favorecer el despegue de las placas de ateroma de las paredes de las arterias y que incorporadas al torrente sanguíneo pueden provocar trombosis e infartos.

#### Figura 4

Ilustración rangos de temperatura para una persona



Nota. Tomado de Temperatura Corporal por *Hermosillo Heart Team HHT*, sf.

### ***3.1.4 Microcontrolador***

Un microcontrolador supone la integración de los componentes necesarios para realizar procesamiento en un único chip (Gridling, G., & Weiss, B., 2007), éste cuenta con un software integrado para su funcionamiento “En su memoria sólo reside un programa destinado a gobernar una aplicación determinada; sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores del dispositivo a controlar, y todos los recursos complementarios disponibles tienen como única finalidad atender sus requerimientos. Una vez programado y configurado el microcontrolador solamente sirve para gobernar la tarea asignada.” (Novas, D. , 2008)

### ***3.1.5 Internet de las cosas (IOT)***

Se ha definido cómo “un nuevo paradigma tecnológico concebido como una red global de máquinas y dispositivos capaces de interactuar entre sí “(Lee, I., & Lee, K. ,2015)

La Internet de las cosas (IoT) describe la red de objetos físicos ("cosas") que llevan incorporados sensores, software y otras tecnologías con el fin de conectarse e intercambiar datos con otros dispositivos y sistemas a través de Internet. Estos dispositivos van desde objetos domésticos comunes hasta herramientas industriales sofisticadas. Mediante la informática de bajo costo, la nube, big data, analítica y tecnologías móviles, las cosas físicas pueden compartir y recopilar datos con una mínima intervención humana. En este mundo hiperconectado, los sistemas digitales pueden grabar, supervisar y ajustar cada interacción entre las cosas conectadas. El mundo físico y el digital van de la mano y cooperan entre sí. (Oracle, s. f.)

### ***3.1.6 Arquitectura de software***

La arquitectura de software refiere el diseño estructural de un producto de software mediante el uso de patrones y reglas que puedan ser utilizadas y replicadas, se ha definido su funcionalidad de la siguiente manera: “Se ocupa de la selección de elementos arquitectónicos, sus interacciones y las restricciones sobre esos elementos y sus interacciones necesarios para proporcionar un marco en el que satisfacer los requisitos y servir de base para el diseño (Perry, D. & Wolf, A., 1992).

### ***3.1.7 Bluetooth de baja energía***

De las siglas BLE (Bluetooth Low Energy) en inglés es una implementación de Bluetooth con funcionalidades orientadas a una baja energía de uso, con diferencias clave frente al bluetooth tradicional. (Bluetooth SIG., s. f.).

### ***3.1.8 Sensor de temperatura de tipo infrarrojo***

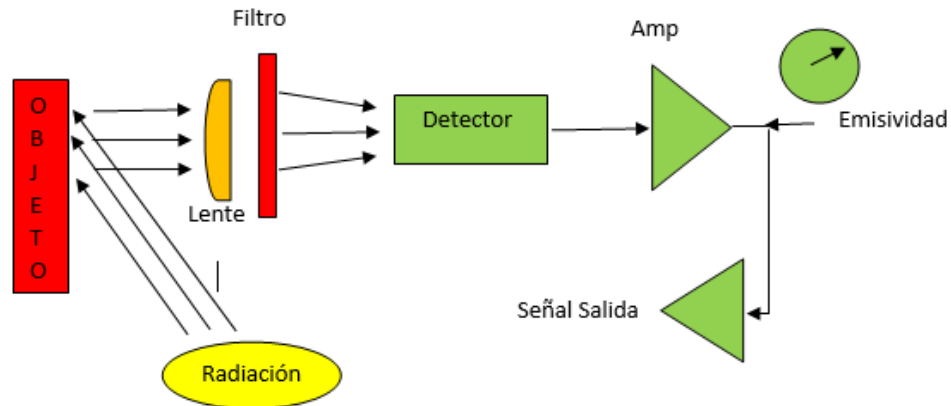
Los sensores infrarrojos son compactos de alta calidad y bajo costo, miden la temperatura de objetos y materiales inaccesibles o en movimiento. Con precisión, consistencia y un sorprendente tiempo de respuesta.

Estos sensores son óptimos para la industria alimenticia, del vidrio, papelera, del plástico, acerera, asfáltica, automotriz y energética; Realizan mediciones de alto rendimiento, miden temperaturas desde -20°C a 2000°C. Son compatibles con gran variedad de instrumentación sin realizar interfaces o acondicionamiento de señal. Existen diversas resoluciones ópticas, se pueden configurar desde una computadora mediante software el cual está incluido. Permite configurar el rango y la emisividad, aplicar filtros, lecturas mínimas, lecturas máximas, medias e instantáneas. Funciona en ambientes de hasta 50°C aunque existe una versión con chasis para

refrigerar con agua y de un collar para salida de aire para condiciones más adversas. (Logicbus S.A. de C.V., s. f.).

**Figura 5**

Esquema de funcionamiento básico de un sensor infrarrojo



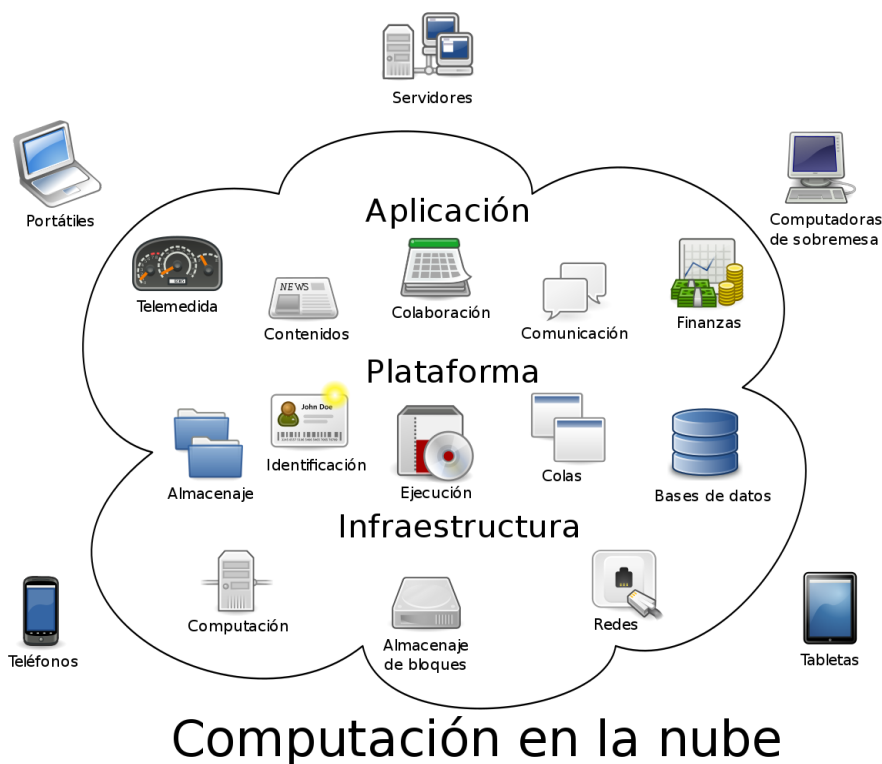
### 3.1.9 Computación en la Nube

La computación en nube es un modelo que permite el acceso ubicuo, cómodo y a disposición a un conjunto de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que pueden ser rápidamente provisionados y liberados con un mínimo esfuerzo de gestión o interacción con el proveedor de servicios (Mell, P., & Grance, T., 2011).

Dentro de los modelos de despliegue existe la nube privada, cuándo la infraestructura de la nube es proveída para uso exclusivo de una organización; la nube comunitaria, que refiere a la infraestructura dada para ser usada por una comunidad de consumidores con características en común; la nube pública, la cuál es proveída para el uso del público en general y finalmente la nube híbrida que puede ser la composición de 2 o más de los modelos mencionados.

**Figura 6**

Esquema de referencia para concepto de computación en la nube



Nota. Imagen tomada de Sam Johnston, con licencia CC BY-SA 3.0. Recuperada de

<https://commons.wikimedia.org/w/index.php?curid=33656837>

### ***3.1.10 Serverless Computing***

La informática sin servidor se define como un modelo de desarrollo y ejecución que permite a los desarrolladores construir y ejecutar aplicaciones sin provisionar o administrar servidores o una infraestructura backend. El proveedor de la nube se encarga del aprovisionamiento de la infraestructura necesaria para ejecutar el código y ampliando y reduciendo la infraestructura según sea necesario. El proveedor de la nube también es responsable de toda la gestión y el mantenimiento rutinario de la infraestructura, como las

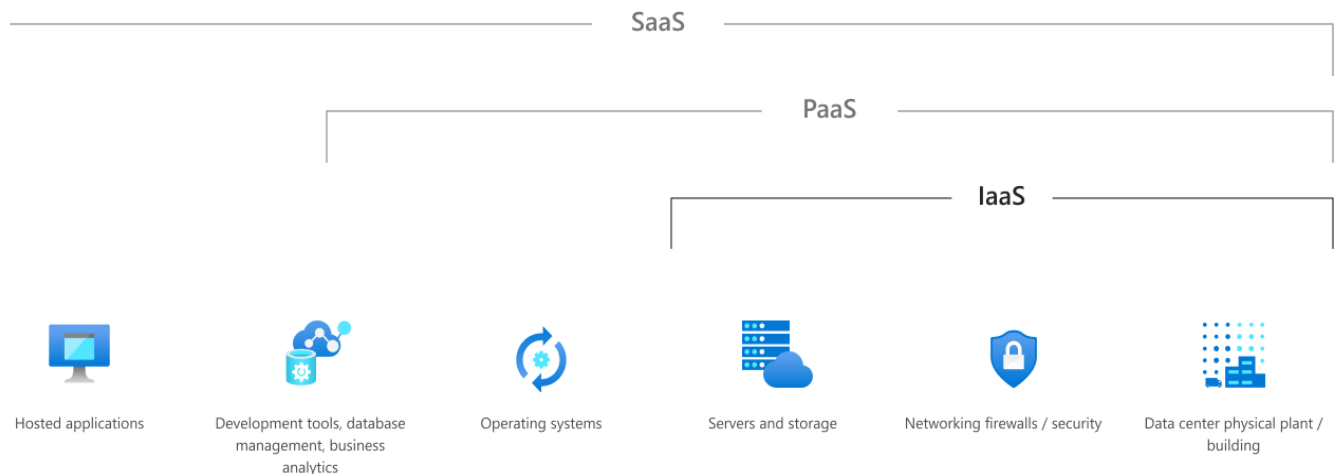
actualizaciones y los parches del sistema operativo, la gestión de la seguridad, la planificación de la capacidad, la supervisión del sistema, etc. (IBM Cloud Education, 2022)

**3.1.11 Infraestructura Como Servicio**

La infraestructura como servicio (IaaS) es un tipo de servicio de informática en la nube que ofrece recursos esenciales de proceso, almacenamiento y redes a petición que son de pago por uso. IaaS es uno de los cuatro tipos de servicios en la nube, junto con el software como servicio (SaaS), la plataforma como servicio (PaaS) y la tecnología sin servidor. (Microsoft Azure, sf)

**Figura 7**

Diagrama distribución de servicios ofrecidos en la nube



Nota. Tomado de IaaS, PaaS & SaaS [Ilustración] por Microsoft Azure, sf.

**3.1.12 KiCad**

Según la definición propia en la página web de la aplicación, esta se trata de una suite de software de código abierto para la automatización del diseño electrónico (EDA). En la cual los programas gestionan la captura de esquemas y el diseño de PCB con salida Gerber.

La suite se ejecuta en Windows, Linux y macOS y tiene licencia GNU GPL v3 (KiCad EDA., 2021)

### ***3.1.13 Peticiones HTTP***

Las peticiones HTTP son mensajes enviados por un cliente, para iniciar una acción en el servidor. Su línea de inicio está formada por tres elementos: método HTTP, Objetivo de la petición (comúnmente una URL) y la versión de HTTP. Adicionalmente cuentan con cabeceras y un cuerpo. (MDN contributors, 2020)

### ***3.1.14 Respuestas HTTP***

Las respuestas HTTP son mensajes dados en respuesta por parte del servidor a raíz de una petición recibida, estas constan de una línea de estado, cabeceras y un cuerpo. Respecto a las peticiones sólo varían en la línea de estado que escribe el resultado de la petición en el servidor, por ejemplo 404 NOT FOUND, si no fue encontrado el recurso, 200 SUCCESS, si la petición fue exitosa, entre otros.

### ***3.1.15 Protocolo HTTPS***

Es un protocolo que permite establecer una conexión segura entre el servidor y el cliente, por lo que en caso de verse interceptada la información no pueda ser leída en claro por personas no deseadas. Se podría definir cómo la versión segura de http (Hypertext Transfer Protocol)

### ***3.1.16 Certificado SSL***

Un certificado SSL es un certificado digital que autentica la identidad de un sitio web y permite conexiones cifradas. SSL significa Secure Sockets Layer, un protocolo de seguridad que crea una conexión cifrada entre un servidor web y un navegador web. Este certificado protege su

conexión a Internet y evita que los delincuentes lean o modifiquen la información enviada entre dos sistemas.

### ***3.1.17 API REST***

Es una interfaz de programación de aplicaciones (API o API web) que se ajusta a los límites de la arquitectura REST y permite la interacción con los servicios web de RESTful (Red Hat, s. f.)

REST no es un protocolo ni un estándar, sino más bien un conjunto de límites de arquitectura. Los desarrolladores de las API pueden implementarlo de distintas maneras, se apoya en HTTP, los verbos que utiliza son exactamente los mismos, con ellos se puede hacer GET, POST, PUT y DELETE.

### ***3.1.18 Librería de programación***

Una librería o biblioteca en programación hace referencia a un conjunto de implementaciones, en código, las cuales ofrecen una interfaz definida para su utilización, las librerías no pretenden ser ejecutadas por sí mismas, sino ser utilizadas a través de otros software.

Entre otras cosas permiten reducir el tiempo de desarrollo, evitar la duplicidad de código y mejorar la calidad del mismo. (Bravo, L., 2018)

### ***3.1.19 React***

React es una librería de Javascript, desarrollada por Facebook y la comunidad de software libre, se trata de un proyecto de código abierto. Esta librería ha sido diseñada con el fin de facilitar el crear interfaces de usuario, específicamente en aplicaciones de una sola página.

## **3.2 Estado del Arte**

### ***3.2.1 Huawei Health***

Aplicación móvil que hace uso de diferentes dispositivos inteligentes o wearables de la marca Huawei, vinculándolos al dispositivo móvil y centralizando la información en esta aplicación, presenta informes de los datos tomados de estos dispositivos (ritmo cardíaco, porcentaje de oxígeno en sangre, peso, entre otros) y permite al usuario de la aplicación visualizarlos y personalizarlos, tiene un propósito de uso personal, dónde sólo el usuario del dispositivo puede acceder localmente. (Huawei, s. f.)

### ***3.2.2 Diseño y construcción de un sistema basado en IoT para supervisión remota de la frecuencia cardíaca y el tono simpático a partir de la señal fotopletoismográfica (PPG).***

Sistema basado en IoT, para la supervisión remota de la frecuencia cardíaca y el tono simpático a partir de la señal fotopletoismográfica (PPG), para la toma de datos se construyó un dispositivo que simula la señal PPG y no se usaron sujetos de prueba y se utiliza la tarjeta ESP32 para realizar la conexión por medio de internet hacia un servidor TCP y posteriormente se visualizan en una aplicación web. (Rodríguez, J. & Argüello, E., 2020)

### ***3.2.3 Monitoreo del ritmo cardíaco a través de dispositivos móviles***

Se presenta un software para dispositivos móviles que hace uso de un dispositivo Zephyr HxM Bluetooth que permite monitorear el ritmo cardíaco mediante una banda puesta en el pecho del paciente, la aplicación desarrollada toma mediciones de la banda y las muestra en pantalla, adicionalmente permite agregar números telefónicos a los cuáles se enviará un SMS en caso de presentarse una anomalía en las mediciones normales, en complemento a la aplicación móvil los

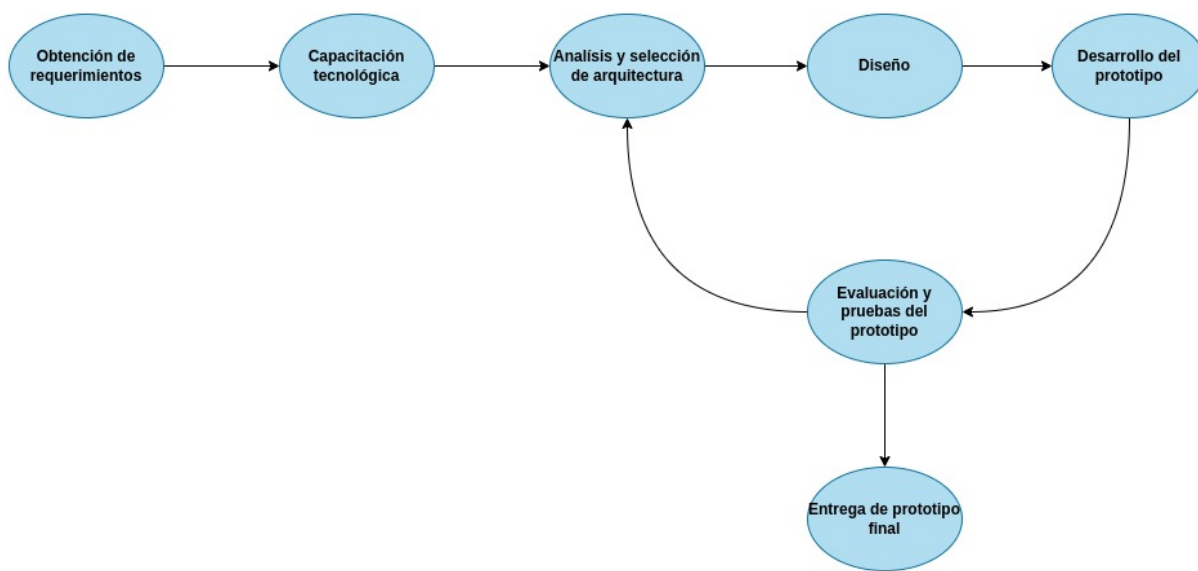
datos se envían a una página web dónde pueden ser consultados. (Aveiga, C. E., Criollo, B. G., & Cruz, A. M., 2018)

#### 4. Marco Metodológico

Se utilizará un enfoque tradicional en el cuál se seguirán las siguientes etapas descritas:

**Figura 8**

Esquema de metodología de trabajo



##### 4.1 Obtención de requerimientos

En la primera etapa de la metodología de trabajo elegida se busca obtener los requerimientos del proyecto los cuales deben definir a nivel funcional lo que se espera del proyecto, así cómo delimitar el alcance de la implementación.

##### 4.2 Capacitación

La ejecución de esta etapa se debe realizar en paralelo a la de análisis y selección de arquitectura puesto que en base a las herramientas necesarias para llevar a cabo la implementación y por

consiguiente realizar una capacitación en el uso de estas herramientas. El propósito principal de esta etapa es suplir a los miembros del proyecto de los conocimientos necesarios para llevar a cabo de forma exitosa el mismo.

#### **4.1 Análisis y selección de arquitectura**

Para esta etapa, en primera instancia se busca identificar cada uno de los componentes a desarrollar, los cuales paso a paso se asocian para finalmente cumplir con el objetivo del proyecto. Una vez identificados estos componentes, se procede a plantear una arquitectura que se ajuste a las condiciones indicadas.

#### **4.2 Diseño**

Si bien esta etapa se puede extender a lo largo del proyecto, la idea es encontrar la mejor ruta de trabajo según los plazos definidos en el cronograma y los recursos con los que se cuenta; también, dependiendo del método de trabajo y planeación durante el desarrollo del proyecto, se van conociendo más datos, consecuencias e ideas de mejora, y así, se hace necesario realimentar acerca del modo de trabajo para optimizar recursos.

#### **4.3 Desarrollo del prototipo**

En esta parte, se desarrollará y validará las actividades propuestas, que en conjunto, deben cumplir con los diferentes objetivos específicos del proyecto.

#### **4.4 Evaluación y pruebas de prototipo**

En el momento en que se logre obtener el objetivo principal del proyecto, se procede a realizar pruebas y evaluar el prototipo. Se requiere comprobar que el proyecto funcione conforme al plan diseñado en los tiempos estimados y con los recursos destinados; dependiendo de esto, se podrá determinar la efectividad, calidad del producto, los riesgos, mejoras a futuro, entre otros factores de evaluación cuantitativa.

#### **4.5 Entrega de prototipo final**

Se hará la entrega del prototipo final una vez el proyecto culmine exitosamente la fase de pruebas y evaluación, cumpliendo en su totalidad con el objetivo general del proyecto.

### **5. Desarrollo del Proyecto**

#### **5.1 Obtención de requerimientos**

Formalmente un cliente realiza la solicitud de un producto software mediante una serie de requerimientos funcionales, describiendo en estos la situación actual, sus objetivos, el resultado esperado, y también delimita el alcance que debe tomar su implementación. Para este caso, obedeciendo a los objetivos del proyecto, se planteó crear un prototipo de plataforma IOT para el monitoreo cardíaco de pacientes bajo cuidados médicos. Esta plataforma debe recibir los datos captados por los sensores, temperatura, oxígeno en sangre y pulso cardíaco, teniendo la capacidad de representarlos en la app móvil y en la plataforma, así como también debe permitir medir dichas variables con la frecuencia que el encargado del paciente considere necesario.

Se debe registrar visualmente el pulso cardíaco mediante la curva que traza la señal capturada por el sensor y se debe almacenar cada registro de medición en un historial donde se podrá ver la secuencia e información de los monitoreos. La plataforma debe permitir configurar parámetros, los cuales serán rangos que indican puntos críticos en las mediciones tomadas; esto funcionará de manera que, cuando el sensor capte las señales y en la plataforma se registren datos superiores o inferiores a los puntos críticos, se deben generar alertas que se puedan visualizar se notifique en la app y en la plataforma, registrando también los valores de mediciones sin que se tenga que hacer manualmente.

La conexión será vía bluetooth desde el dispositivo móvil al microcontrolador y la plataforma estará alojada en la nube, de manera que en su interfaz se puedan visualizar usuarios, y se permita configurar recursos, servicios y demás componentes de manera ágil sin necesidad de depender de la administración de un servidor. La aplicación debe contar con un logueo de usuario y los registros en la plataforma deben poderse filtrar fácilmente por nombres y fecha de creación del registro.

## **5.2 Capacitación**

Antes de plantear la arquitectura, hay que evaluar y definir qué herramientas y conocimientos permiten el desarrollo de los requerimientos anteriormente mencionados. Siendo así, se opta por profundizar en diversas temáticas para reforzar el conocimiento y predecir con mayor certeza qué tan viable es desarrollar el sistema con cada una de las diferentes opciones consultadas. En primera instancia y en paralelo con las definiciones de arquitectura se ve necesario adquirir conocimiento sobre los servicios ofrecidos por proveedores en la nube, debido a que los miembros del proyectos contaban con algunos conocimientos, se evaluaron estos y se determinó que la mayoría de conocimientos estaban en torno a AWS, situación que influyó en la selección de arquitectura.

A medida que se obtenía información acerca de la implementación de la arquitectura serverless con AWS, se identificaron una serie de recursos que también podrían asociarse a la implementación de la plataforma y que significa un ahorro de tiempo, de código y de interconectar diversas tecnologías para lograr exactamente lo mismo, y con esto se está haciendo referencia a Amplify de AWS, una herramienta que permite el desarrollo simplificado de aplicaciones web y móviles, proporcionando backend, frontend, autenticación, almacenamiento, bibliotecas, entre otros recursos que se pueden gestionar. Mediante documentación que provee la

propia plataforma de AWS, así como guías o documentos generados por la comunidad. Adicionalmente se realizaron cursos en plataformas como Coursera y LinkedIn Learning. A partir de estos conocimientos adquiridos se pudieron realizar ejercicios y primeras implementaciones relacionadas con las necesidades del proyecto que permitieron generar así una mayor profundidad en la formación en AWS y demás.

### **5.3 Análisis y selección de arquitectura**

#### ***5.3.1 Análisis de alcance, viabilidad y riesgos***

Partiendo de la definición de alcance dada, donde se requiere una plataforma que permita captar, procesar y mostrar los datos vitales de pacientes, sin que estos últimos se encuentren dentro de una instalación médica. Significando entonces que deben contar con una conexión de algún tipo para la transmisión de estos datos, así mismo el dispositivo debe ser cómodo para el uso del paciente.

Entendiendo también que se trata de un prototipo sujeto a cumplir con los mínimos requerimientos identificados y limitado por los recursos con los que se cuenta, buscando sólo realizar pruebas a pequeña escala.

Se identifica entonces la viabilidad del proyecto en torno a que el alcance dado acata las limitaciones y establece objetivos mínimos a cumplir dentro del proceso de realización del proyecto. Con esto se identifican distintos patrones viables de arquitectura que pudiesen suplir la definición inicial.

Entendiendo la naturaleza del proyecto que se mueve alrededor de los datos. Se establecen entonces riesgos que pudiesen comprometer estos, cómo puede ser la pérdida de acceso a estos datos, la fuga a terceros de los mismos, o la pérdida total de estos.

El diagrama de casos de uso presentado en la figura 9 representa las actividades principales con las que debe cumplir la plataforma, ofreciendo al usuario interactuar con cada una de ellas, las cuales cumplen con funcionalidades específicas esenciales para realizar el flujo básico del monitoreo de signos vitales y lanzamiento de alertas.

### Figura 9

Diagrama de casos de uso



#### 5.3.2 Identificación de componentes y factores

Con el fin de lograr el desarrollo de una plataforma que cumpla con estándares de calidad y optimización, se plantea mediante el uso de nuevas herramientas en vista de conseguir una arquitectura contemporánea e innovadora que cumpla con las siguientes condiciones:

- El dispositivo debe poder conectarse con los servicios de cómputo y comunicarse con estos para el envío y recepción de datos.
- Los datos y manejo de persistencia se pueden acceder directamente desde la web.
- La base de datos usada puede ser no relacional, bajo la estructura de llave-valor.
- La lógica, algoritmos y validaciones pueden ser proveídos por servicios de terceros.
- Servicios de infraestructura como autenticación pueden ser resueltos por terceros.
- Fraccionar el servidor en servicios
- El flujo y los aspectos de lógica de la aplicación son coordinados no por el servidor sino por el cliente.
- Independencia entre lógica y vista, donde ésta consuma los recursos de alojamiento en el usuario.
- La gestión del ciclo de vida y escalabilidad de la aplicación puede ser garantizada por terceros.

### ***5.3.3 Evaluación de condiciones y límites para desarrollo de los componentes***

Para el desarrollo de los componentes previamente establecidos se debe tomar en cuenta la necesidad de capacidad de cómputo, almacenamiento y comunicación para estos, es entonces donde se plantean alternativas para suplir esta necesidad, entre un servidor propio, recursos de computación en la nube o una solución híbrida, entendiendo también las limitaciones de recursos existentes, tanto económicos como intelectuales, pues algunas de estas soluciones requieren un conocimiento específico para su implementación. Adicionalmente se debe tener en cuenta la ergonomía del dispositivo, su facilidad de conexión y su costo.

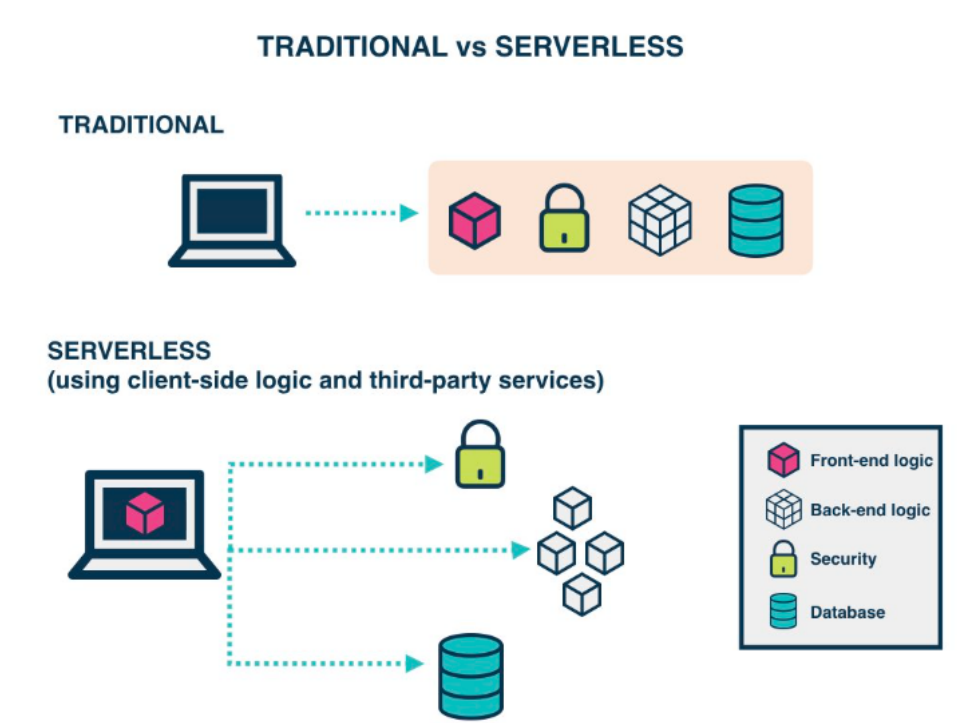
### ***5.3.4 Planteamiento de arquitectura***

Una vez finalizados los pasos anteriores se tienen definiciones y limitaciones para el planteamiento de la arquitectura que pueda suplir los requerimientos identificados. Con esto se llega entonces a la conclusión de que es necesario usar una arquitectura orientada a la web.

Surgen entonces las opciones de usar una arquitectura serverless (sin servidor) y la tradicional basada en un servidor propio, ya sea alojado o local, o la opción de una arquitectura híbrida; sin embargo, para ajustar la arquitectura a los los requerimientos definidos, donde se menciona que no hay interés alguno en manipular un servidor físico, sino que este debe estar alojado en la nube, por lo tanto, la arquitectura debe ser una serverless.

### Figura 10

Comparación entre arquitectura tradicional y sin servidor



Nota. Tomado de Traditional vs Serverless por R. Weston, 2017

Finalmente decantándose por la arquitectura serverless, pues es esta quién mejor se ajusta a las pautas establecidas. Teniendo en cuenta esto se exploraron diferentes proveedores de cloud, los cuáles pudiesen suplir las necesidades:

**5.3.4.1 Amazon Web Services (AWS).** Los servicios de AWS representan un candidato importante para la implementación de la arquitectura diseñada, puesto que cuenta con una base no relacional (Amazon DynamoDB) basada en clave-valor, así mismo cuenta con capacidad de procesamiento serverless (Funciones Lambda) que permiten ser ejecutadas únicamente cuándo sean llamadas, y sólo generarán un costo en este momento, también cuenta con un servicio de gateway para direccionar y gestionar peticiones HTTP (Amazon API Gateway), también cuenta con un servicio para gestión de usuarios (Amazon Cognito) y sobretodo cuenta con un servicio para gestión de aplicaciones serverless (Amazon Amplify). En complemento con esto se tiene el hecho de que los miembros del proyecto cuentan con conocimientos previos en AWS, además de experiencia con implementaciones similares y el uso de los servicios mencionados.

### **Figura 11**

Tabla de precios implementación de componentes en la nube AWS

The screenshot displays the AWS Pricing Calculator for a service named 'HeartMonitor'. The 'Estimate summary' section shows the following costs:

- Upfront cost: 0.00 USD
- Monthly cost: 8.99 USD
- Total 12 months cost: **107.82 USD** (Includes upfront cost)

The 'My Estimate' section contains a table of services:

<input type="checkbox"/>	Service Name		Upfront cost	Monthly cost
<input type="checkbox"/>	AWS Amplify	<a href="#">Info</a>	0.00 USD	2.70 USD
<input type="checkbox"/>	AWS Lambda	<a href="#">Info</a>	0.00 USD	0.00 USD
<input type="checkbox"/>	Amazon API Gateway	<a href="#">Info</a>	0.00 USD	0.00 USD
<input type="checkbox"/>	Amazon Cognito	<a href="#">Info</a>	0.00 USD	5.00 USD
<input type="checkbox"/>	Amazon DynamoDB	<a href="#">Info</a>	0.00 USD	1.29 USD

Nota. Adaptado de Consola de administración de AWS. por Amazon Web Services, (sf).

Se usaron componentes que suplan necesidades previamente definidas, con una demanda de uso estimado para unos 100 usuarios mensuales.

**5.3.4.2 Google Cloud Platform.** Los servicios de GCP cuenta con una base no relacional (Google Cloud Datastore) basada en clave-valor, en cuanto a capacidad de procesamiento serverless existe una alternativa de funciones (Cloud Functions) que sólo generarán un consumo de recursos y por ende un costo en el momento en que estas reciban una petición, para el manejo de las peticiones HTTP cuenta con su propio API Gateway, en torno a la gestión de usuarios tiene un servicio específico (Identity Platform), Google Cloud no cuenta con un servicio específico para el manejo de aplicaciones serverless, sin embargo mediante Google Firebase se pueden integrar servicios en torno a una aplicación.

En revisión a los recursos técnicos de los miembros del proyecto vemos que se cuenta con algunos conocimientos en Google Cloud Platform, sin embargo desconocen implementaciones en la mayoría de los servicios mencionados.

### **Figura 12**

Tabla de precios implementación de componentes en la nube GCP

Estimate

Datastore ✎ ✖

---

Stored data: 1 GiB

---

Entity Reads: 10,000,000

---

Entity Writes: 100,000

---

**USD 5.09**

---

A portion of your estimate fits within the [Datastore free tier](#).

Cloud Functions ✎ ✖

---

Region: Iowa

---

Invocations: 100,000	USD 0.00
RAM (GiB-seconds): 1,475 per month	USD 0.00
CPU (GHz-seconds): 2,360 per month	USD 0.00
Networking: 48.828 GiB per month	USD 5.26

---

**USD 5.26**

---

A portion of your estimate fits within the [Cloud Functions free tier](#).

Identity Platform ✎ ✖

---

MAU from Social, Anonymous, Email+Password and Phone : 100

---

MAU from Enterprise Federation Providers (SAML and OpenId Connect): 0

---

Successful Verifications with SMS to United States, Canada and India: 0

---

Successful Verifications with SMS to countries outside United States, Canada and India: 100

---

**USD 0.00**

---

A portion of your estimate fits within the Identity Platform free tier. Please [click here](#) for more detail.

**Total Estimated Cost: USD 10.35 per 1 month**

Estimate Currency

USD - US Dollar ▼

EMAIL

SAVE

DOWNLOAD\*

Nota. Adaptado de Google Cloud Pricing Calculator. por Google Cloud. (sf).

Se usaron componentes que suplen las necesidades previamente definidas, con una demanda equivalente a la usada para AWS.

Realizando una comparación entre lo ofrecido por GCP y AWS tenemos algunos datos que nos ayudan a definir cuál de estas es más favorable para el desarrollo del proyecto, esta comparación la podemos observar en la tabla 1.

**Tabla 1**

Precios de los proveedores de servicios en la nube Amazon Web Service (AWS) y Google Cloud Platform (GCP)

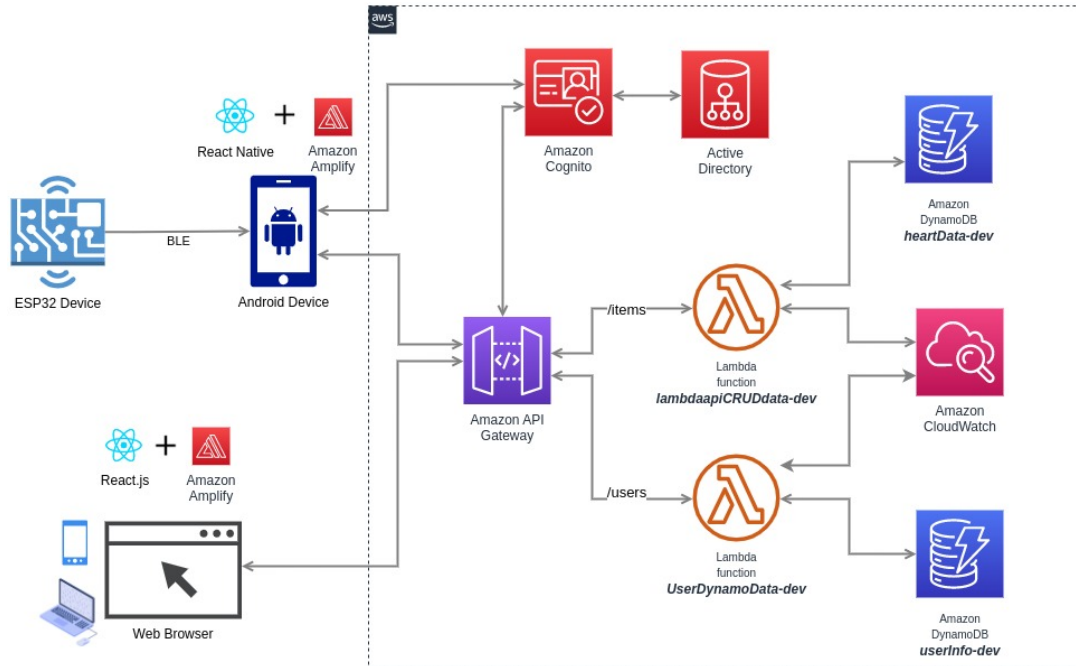
Servicio	AWS		GCP	
	Nombre	Precio (USD)	Nombre	Precio (USD)
Cómputo Serverless	Lambda Functions y API Gateway	0	Cloud Functions	5.26
Almacenamiento persistente	DynamoDB	1.29	Datastore	5.09
Autenticación	Cognito	5	Identity Platform	0
Administración de aplicaciones	Amplify	2.7	Firebase	0
Total mensual	8.99		10.35	
Total anual	107.82		124.2	

Nota. Para estas estimaciones se usaron los mismos valores mensuales, de 100000 peticiones, 10000000 lecturas al almacenamiento y 100000 escrituras, junto con 100 usuarios en el directorio activo.

Se opta entonces por una arquitectura basada en la nube, con los servicios de AWS como núcleo de esta.

**Figura 13**

Diagrama de arquitectura, solución serverless basada en los servicios de AWS



Nota. El diagrama de arquitectura representa la construcción de la plataforma en alto nivel.

## 5.4 Diseño

### 5.4.1 Realización de estimaciones

Es necesario contar con proyecciones de tiempo para lograr una efectiva división de tareas y coordinar el desarrollo del proyecto a medida que se logra la capacitación, con el fin de lograr una entrega funcional de la plataforma prototipo consistente e integral.

Tomando en cuenta que ya hay una arquitectura definida, se debe investigar acerca del desarrollo de aplicaciones serverless y cada uno del desarrollo e implementación de sus componentes, para lo cual se tiene destinado aproximadamente 30 días, dentro de los cuales se va a desarrollar paso a paso conjuntamente con el montaje del microcontrolador, del cual ya se estaría realizando su implementación gracias a la capacitación previa sobre el tema.

La definición de requerimientos, dependencias, diagramas de procesos y vistas de usuario necesarios para lograr una proyección funcional, se considera que toma alrededor de 10 días, de los cuales, se trabaja inicialmente con la captación de datos del prototipo hardware, y de esta manera definir la utilización y representación de datos en la plataforma.

Una vez evidenciada la toma de datos de manera consistente y representativa de cada una de las variables medidas, se lleva a cabo el montaje de servicios y peticiones en la nube tal cual se define en la arquitectura serverless, junto con la implementación de alojamiento de datos, autenticación, manejo de los datos e interacción con interfaz gráfica, comprobando la efectiva conexión e interacción entre el dispositivo hardware y el componente base de software.

Posterior a las efectivas pruebas de conexión entre el hardware y software, se procede a realizar el seguimiento de los datos arrojados, a su vez, realizando los respectivos ajustes, mediciones e implementando las funcionalidades necesarias para el análisis del que dependen las alertas y acciones de monitoreo en tiempo real; para la efectividad de esta sección se estima alrededor de 15 días.

Una vez comprobado el éxito funcional de la aplicación, se realiza una fase de pruebas que consistiría en la aplicación del monitoreo efectivo en un grupo reducido de personas voluntarias, de las cuales se toman muestras de asertividad, aceptación y sugerencias para la realimentación y embellecimiento de la plataforma; lo cual tomaría aproximadamente 5 días.

Para finalizar, se realizan las correcciones necesarias front-end para facilitar la interacción con el usuario, y se integra el dispositivo físicamente para brindar mayor comodidad de usabilidad, de esta manera, se configura el sistema para estabilizarse, calibrarse, y así estar preparado para la presentación final en los próximos días.

#### 5.4.2 Identificación y asignación de recursos

En primera instancia, para realizar el montaje básico hardware, se hace necesario la compra de una placa base para dicha instalación, en conjunto con el circuito base para provisión de energía, microcontrolador ESP-32 con sus respectivos sensores de temperatura, oxígeno en sangre y pulso cardiaco, que en total por un valor de \$120.500, según los precios obtenidos en mercado libre, a la fecha del 25 de enero de 2023, precios que pueden variar en el tiempo y según el proveedor.

**Tabla 2**

Componentes previstos para la construcción del prototipo con su precio actual.

<b>Dispositivo</b>	<b>Valor</b>
Tarjeta ESP32	34000
Sensor De Concentración De Oxígeno Max30102	25000
Mlx90614 Sensor De Temperatura Infrarrojo Para Arduino	43000
Display Pantalla Lcd Oled I2c 0.96 128*64 7 Pines Arduino	18500
<b>Total</b>	120500

Para la proyección de la interfaz de usuario y como objeto de pruebas se puede usar un teléfono celular de sistema operativo Android para asegurar la compatibilidad con el aplicativo y el sistema hardware; por otra parte, para el desarrollo del aplicativo y parte pruebas, se hace necesario la disponibilidad de un computador, los cuales no representan costo por su previa adquisición.

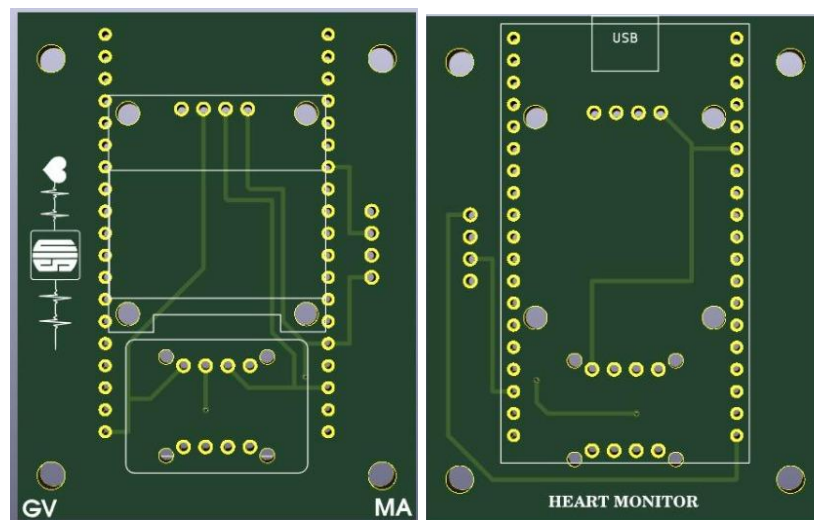
### 5.4.3 Establecimiento de ruta de trabajo

De acuerdo a la metodología escogida y siguiendo la ruta planteada que se propone en esta, la arquitectura se desarrolla principalmente bajo el concepto de diseñar un aplicativo serverless, dado que así se identifican las posibles herramientas para el desarrollo software de esta partiendo del backend para llegar finalmente a implementar un frontend interactivo con el usuario. Habiendo conocido las herramientas y los servicios que ofrecen estas, se hace la repartición de tareas para cada entorno de desarrollo y se revisa que tengan la compatibilidad necesaria entre ellos para conformar un software robusto y de calidad, así como también su óptimo funcionamiento predecible para captar las señales del microcontrolador en tiempo real.

Así como se diseña una interfaz de usuario software para la plataforma, también se realiza el boceto para imprimir la placa en que se hará el montaje hardware, lo cual tendría un aspecto acorde a las figura 14, diseñada en la aplicación KiCad.

**Figura 14**

Diseño placa PCB para montaje de circuito



Nota. Se trata de una visualización en 3D de una placa PCB diseñada en KiCad

Sobre dicha placa se realiza el montaje de los componentes electrónicos y se verifica su funcionamiento inicialmente comprobando el paso de energía y procediendo a probar los datos de lecturas arrojados por el ESP-32 Dev Kit.

Con los programas establecidos para el desarrollo e implementación de código back-end, front-end y consumos de servicios, se inicia un prototipo base para comprobar la percepción de señales, aún sin analizar tales datos. Cabe resaltar que después de esto, el enfoque se centra en la obtención de alertas y cómo hallar estas señales una vez las 3 variables son captadas por el dispositivo.

A medida que se realizan pruebas, se va evidenciando el éxito del desarrollo y el nivel de certeza frente a posibles situaciones de emergencia; esta fase se lleva a cabo en acompañamiento de diversos voluntarios para tomar medidas correctivas dentro del aplicativo a tiempo y evaluar riesgos o futuras fallas y opciones de mejora en el software. De igual manera, gracias al acompañamiento de los tutores del proyecto, se contaría con su previa aprobación para realizar estas pruebas de manera adecuada.

En el momento en el que se obtienen los resultados deseados tanto en las funcionalidades que se plantean en los requerimientos como en la utilidad hardware, se puede decir que el prototipo está listo para presentarse y usar como un primer acercamiento a lo que sería una ayuda de monitoreo de signos vitales fuera de instalaciones médicas.

#### ***5.4.4 Análisis de datos y optimización de modo de trabajo***

Teniendo en cuenta que las tres variables consideradas en este proyecto, tienen una razón exacta para ser parte esencial en la generación de alertas en la plataforma, se podría notar que, los datos arrojados por la medición de cada una de las variables son en cierta medida

independientes, aunque esto no significa que las alertas serán generadas por la alteración de una única variable; por el contrario, se necesita de un análisis donde se alcancen los niveles de riesgo límite de cada signo vital, cuando alguno de ellos ha variado drásticamente y poco proporcional a los niveles de las demás variables.

Un cuestionamiento surge en particular, ¿qué variable desarrollar primero?, si lo que se va a implementar radica principalmente en el monitoreo cardiovascular, significa que este debe ser el punto de partida, la captación de datos de los latidos del corazón; esto para iniciar seguramente con una de las partes base y de mayor dificultad, lo cual se puede probar de manera aislada, analizar los resultados, comprobar correcto funcionamiento de hardware y proyección en el software, y una vez realizado este punto, se remontan las otras dos variables, su respectiva instalación, programación para recepción de datos, análisis, y una vez todo marche bien de manera independiente, se calcula cuantitativamente la relación entre los signos vitales, para proyectar posteriormente el punto o los puntos donde el riesgo debe alertar al usuario de la plataforma.

La optimización en términos del desarrollo software, para este caso estaría ligado y limitado a la arquitectura elegida, ya que es orientada a la creación de aplicación serverless, lo que hace que en parte se deba trabajar como una estructura particionada, donde cada una de las partes software back-end e inclusive front-end sean independientes y cumplan sus funciones, siempre y cuando estén alojadas en el hospedador y sintonizadas entre sí, en el momento en que se solicita un consumo en la plataforma. El sólo hecho de usar una arquitectura basada en serverless, aplica diversas ventajas (u optimizaciones) en comparación con las arquitecturas convencionales, tanto por parte del desarrollador, como del consumidor final; también gracias a que esta idea se puede convertir en servicios tecnológicos basados en la nube; AWS Lambda, por

ejemplo, permite que el desarrollo se traduzca a una gran experiencia para el usuario final, facilitando la consulta de histórico del monitoreo del paciente, así como también, para el desarrollador significa mucho menos tiempo en labores de soporte o mantenimiento del servidor.

## **5.5 Desarrollo del prototipo**

### ***5.5.1 Establecimiento del entorno de trabajo***

El entorno de trabajo adecuado para este proyecto, no necesariamente implica el uso o accesibilidad a laboratorios o sitios con algún propósito en específico, ya que la mayoría se obtiene a través de la investigación por medio del computador; de la misma manera en que se lleva a cabo el desarrollo del prototipo. No es necesario el uso de maquinaria ni de implementos más que los que se encuentran asociados a las extensiones y conexiones del microcontrolador, que por otra parte, también es posible simular en aplicativos como KiCad.

Cualquier espacio o superficie donde esté adecuada la instalación del computador y que enseguida se pueda ubicar los implementos hardware, es suficiente para decir que el entorno físico es el adecuado.

Una parte muy valiosa que también hace parte del entorno de trabajo de manera intangible, es el apoyo ya sea presencial o a distancia, síncrono o asíncrono del docente director de proyecto José Rubiano y la dra. Laura Valencia como co-tutora; orientadores que por su experiencia, conocen cómo es el trayecto que se debe ejecutar durante este desarrollo y qué es lo mejor que se puede aportar al proyecto para ofrecer los mejores resultados.

### ***5.5.2 Asignación de tareas***

Para la realización de los objetivos propuestos y el alcance de los requerimientos definidos, se identificaron algunas tareas principales que se definieron cómo:

**5.5.2.1 Búsqueda y selección de los componentes de hardware.** Esta tarea refiere a la consulta en diferentes fuentes, identificación de características y comparación de diferentes componentes de hardware que puedan suplir las necesidades. En esta tarea se evaluará la capacidad, la facilidad de acceso, la compatibilidad, la dificultad técnica de su utilización y algunos otros puntos de significancia para la elección de los componentes.

**5.5.2.2 Adquisición y prueba de los componentes hardware.** Una vez se han definido los componentes hardware, se buscará diferentes opciones para la adquisición de los mismos, y posterior a esto, realizar pequeñas implementaciones que permitan conocer el funcionamiento de los componentes y la capacidad técnica que se requiere para su implementación.

**5.5.2.3 Implementación y prueba del entorno base en la nube.** Para suplir con la arquitectura previamente planteada se deben realizar algunas primeras implementaciones en la nube que permitan identificar el funcionamiento de los diferentes servicios, así como también identificar posibles contratiempos o bloqueantes que surjan a partir de estos. También buscar la forma en que los servicios serán aprovechados de la mejor manera.

**5.5.2.4 Selección y pruebas iniciales de la tecnología front-end.** En esta tarea se explorarán de manera superficial las diferentes tecnologías front-end tanto para móvil cómo web, que permitan una interfaz amigable, una velocidad de desarrollo adecuada, que cuente con una

documentación extensa y librerías necesarias para el desarrollo del proyecto. También se debe tomar en cuenta la viabilidad técnica (curva de aprendizaje, conocimientos previos, etc.)

**5.5.2.5 Montaje inicial de componentes hardware en placa de pruebas.** Por medio de una protoboard se realizará un montaje inicial de los componentes adquiridos que permita realizar pruebas sobre el circuito, identificar las conexiones necesarias, los posibles fallos, suplemento de energía, entre otras cosas.

**5.5.2.6 Desarrollo y pruebas de adquisición de datos.** En este punto se iniciará con el desarrollo de código necesario para realizar la adquisición de datos desde el dispositivo hardware y con esto realizar las primeras pruebas de esta.

**5.5.2.7 Diseño y desarrollo front-end.** Para esta tarea se establecerá el diseño inicial que tendrá la aplicación móvil y la página web, y se iniciará con el desarrollo de estos componentes front-end

**5.5.2.8 Exploración e integración de servicios en la nube con front-end.** Posterior al diseño inicial del front-end se debe realizar la conexión entre los componentes de este con los servicios en la nube, para esto se debe entonces definir el protocolo a usar, entre REST o SOAP e iniciar con las peticiones a los servicios en la nube y probar su funcionamiento.

**5.5.2.9 Definición y desarrollo de modo de conexión entre el dispositivo hardware y la aplicación móvil para transmisión de datos.** En esta tarea se debe realizar la exploración de

diferentes métodos de conexión entre la aplicación móvil y el dispositivo hardware, ver ventajas y desventajas y elegir el más adecuado.

Una vez establecido se debe iniciar con el desarrollo desde el dispositivo y la app móvil para sincronizar su conexión.

**5.5.2.10 Integración completa entre dispositivo hardware, aplicación móvil, servicios en la nube y web.** Una vez se ha realizado la integración de componentes front-end y nube y aplicación móvil y dispositivo hardware se debe realizar la integración de los 3 componentes donde los datos sean adquiridos por el dispositivo, lleguen a la aplicación móvil y de allí sean enviados a la nube para posteriormente ser consumidos.

**5.5.2.11 Pruebas de funcionamiento a nivel de desarrollo y corrección de errores.** Se deben realizar pruebas simples de funcionamiento dónde se garantice un correcto funcionamiento mínimo de la plataforma y corregir posibles errores en el funcionamiento de la misma, ya sea a nivel funcional o no funcional.

**5.5.2.12 Mejoras, optimización y refactor general del código.** Una vez se ha alcanzado un mínimo funcionamiento correcto de la plataforma se realizarán mejoras identificables sobre el código existente, mejorando el tiempo de ejecución de tareas, legibilidad del código, buenas prácticas, entre otras cosas.

### **5.5.3 Ejecución de tareas**

**5.5.3.1 Búsqueda y selección de los componentes de hardware.** Para cumplir con el objetivo de este proyecto, es necesario principalmente orientar todo aquello que se construye en el hardware y que se usaría en software, orientados a IoT, por eso, luego de familiarizarse con los tipos y referencias de microcontroladores, se llegó a la conclusión de que es adecuado y accesible el dispositivo microcontrolador ESP32 de la empresa Expressif Systems, que por sus características como wifi y bluetooth, favorecen el desarrollo de un proyecto basado en IoT, a un bajo costo de energía, y así, se podría realizar un proyecto que intercambie información a la nube y en simultáneo administre datos de un sensor de forma precisa, gracias a su microprocesador de 32-bits.

Para realizar las mediciones de ritmo cardiaco y oxígeno en sangre, se encontró que el Sensor de concentración de oxígeno y ritmo cardíaco MAX30102 combina LEDs, un fotodetector, óptica optimizada y procesamiento de bajo ruido para detectar pulsioximetría y ritmo cardiaco, el cual es compatible con plataformas embebidas, es decir, es apto para aliarse con el microcontrolador escogido.

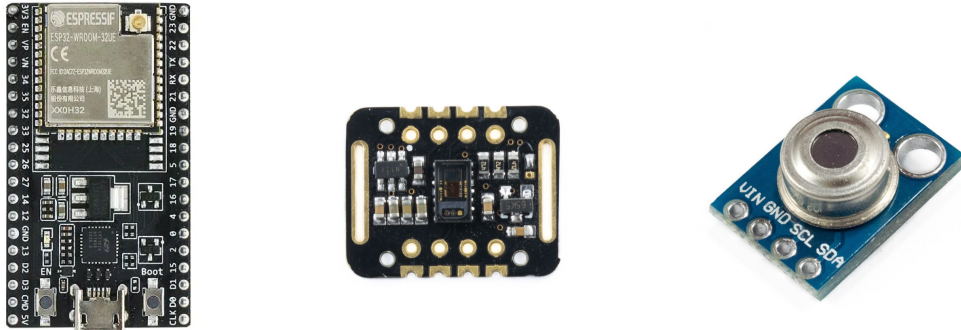
Por último, para la detección de temperatura y captación de dichos datos, se opta por trabajar con el sensor de temperatura infrarrojo MLX90614 fabricado por la empresa Melexis, el cual permite medir la temperatura de un objeto a distancia, es decir, sin contacto físico directo con el dispositivo. Este sensor ofrece la amplificación y digitalización de la señal procedente de la membrana y adicionalmente, la salida del sensor es lineal y se compensa de acuerdo a las variaciones de la temperatura ambiente; esto quiere decir que proveerá un confiable nivel de certeza en los datos proyectados para analizar.

Finalmente, se diseña la placa para realizar el montaje de todos los componentes con sus respectivas conexiones y en el que se pueda mostrar como un dispositivo compacto, integrado y cómodo de usar, claramente, teniendo en cuenta que es un prototipo.

**5.5.3.2 Adquisición y prueba de los componentes hardware.** Los componentes descritos en el punto anterior fueron adquiridos con un tiempo de espera de alrededor de 3 días, siendo así, fué verificada la condición inicial y se comprueba que las referencias eran las solicitadas. Los dispositivos tienen el siguiente aspecto de manera individual.

### Figura 15

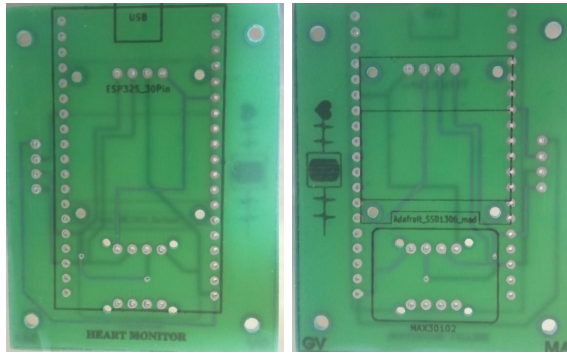
Fotografías de componentes utilizados para realización de dispositivo hardware



Nota. Las imágenes corresponden a los componentes ESP32, MAX30102 y MLX90614 respectivamente, al momento que fueron adquiridos

### Figura 16

Fotografía placa PCB fabricada



Para probar el funcionamiento de estos dispositivos, en primera instancia, se realizaron pruebas simples con el microcontrolador, el cual se conectó a una fuente de 5 voltios y se verificó que estuviera encendido, con paso de corriente y con la ayuda del Arduino IDE, que en este caso, es el entorno escogido para llevar a cabo la programación del microcontrolador, se compilaron códigos simples encontrados en internet para probar las señales generadas en los pines y que a su vez, estuviese funcionando el bluetooth y el wifi. El proceso fué exitoso.

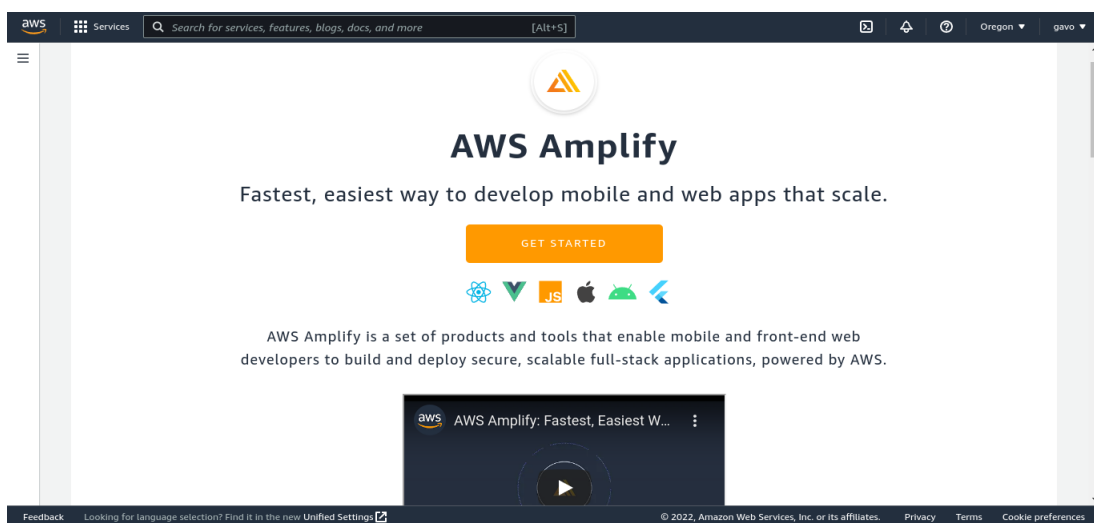
Una vez evidenciado el funcionamiento del ESP32, se consulta en internet las maneras de recibir datos desde los sensores MAX30102 y MLX90614 para ser enviados al microcontrolador. Luego de realizar la conexión y el ajuste de los dispositivos, se configuran funciones básicas para probar los sensores, y se comprueba que efectivamente el microcontrolador lee los datos captados por estos.

Posteriormente se trabaja en conjunto con estos dispositivos y se integrarán en la placa diseñada, de manera que cumplan una función como un sistema en que cada cual aporta sus datos percibidos y el microcontrolador se encarga de procesarlos para convertirlos en información útil para el usuario.

**5.5.2.3 Implementación y prueba del entorno base en la nube.** Con la necesidad existente de crear un entorno en la nube se procede a explorar los diferentes servicios en la nube, dónde se inicia por el servicio de AWS Amplify, pues es este quién nos sirve cómo integrador de los diferentes servicios para una única aplicación serverless.

### Figura 17

Panel de manejo principal del servicio AWS Amplify



Nota. Adaptado de Consola de administración de AWS. por Amazon Web Services, (sf).

Se identificó que el servicio de Amplify permite integración sobre aplicaciones desarrolladas en diferentes tecnologías (React, React Native, Vue, Angular, Android, iOS, etc), así cómo servicios de autenticación, almacenamiento, bases de datos, entre otros, de AWS, dónde nos permite configurar los permisos de acceso y facilitar la interacción entre estos.

Se exploró entonces los métodos de autenticación, en el cuál se eligió cognito, dónde nos permite la administración usuario, elegir requerimientos de contraseña, segundo factor de autenticación, datos requeridos, entre otras cosas.

Para las funciones lambda, cómo solución de cómputo serverless se aprovechó el hecho de que uno de los miembros del equipo había hecho uso previamente de estas, para hacer pruebas de su funcionamiento siendo capaces de ejecutar segmentos de código bajo demanda y manejar peticiones HTTP, generando una respuesta para las mismas.

### Figura 18

Bloque de código correspondiente a una función lambda implementada en node.js

```

/*****
 * HTTP Get method for list all objects *
 *****/
app.get(path, function (req, res) {
  let queryParams = {
    TableName: tableName,
    limit: 100,
  };
  dynamodb.scan(queryParams, (err, data) => {
    if (err) {
      res.statusCode = 500;
      res.json({error: 'Could not load items: ' + err});
    } else {
      res.json(data.Items);
    }
  });
});

/*****
 * HTTP Get method for list objects *
 *****/

app.get(path + hashKeyPath, function (req, res) {
  var condition = {};
  condition[partitionKeyName] = {
    ComparisonOperator: 'EQ',
  };

  if (userIdPresent && req.apiGateway) {
    condition[partitionKeyName]['AttributeValueList'] = [
      req.apiGateway.event.requestContext.identity.cognitoIdentityId || UNAUTH,
    ];
  }
});

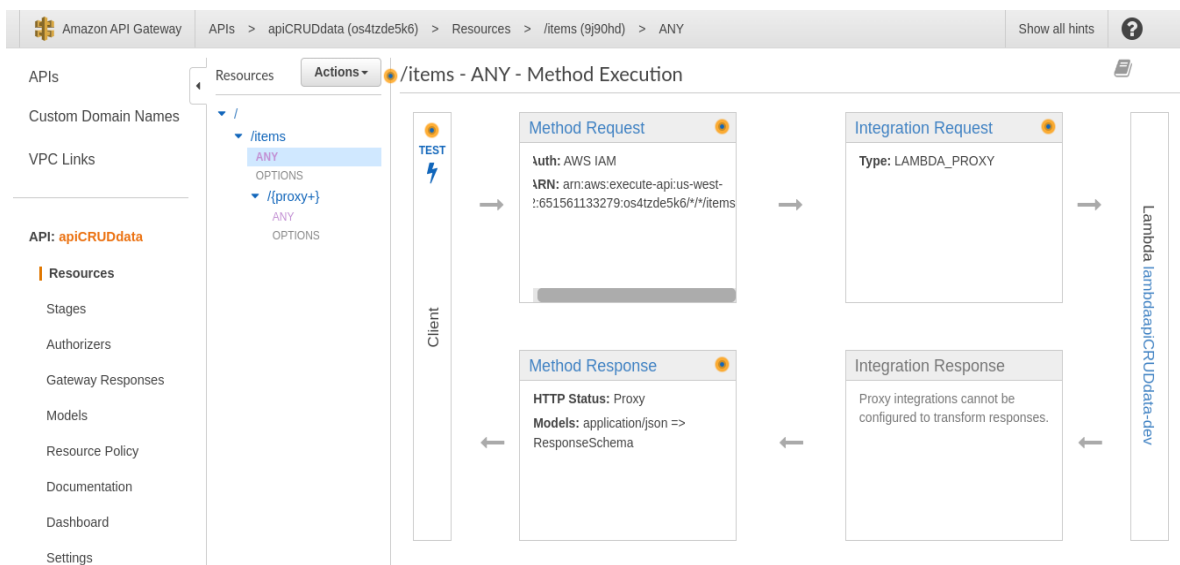
```

Habiéndose optado por utilizar una base de datos no relacional se utiliza DynamoDB que ofrece persistencia y la posibilidad de integración con herramientas cómo amplify permitiendo administrar, por ejemplo el acceso de la función lambda a estos datos, y así mismo contando con diferentes aspectos que permiten salvaguardar los datos.

Para la administración (manejo, redireccionamiento, aceptación, etc.) se hace uso del API Gateway que provee Amazon donde se reciben las peticiones, requiriendo autenticación previamente y se hace el redireccionamiento de estos hacia la función lambda.

**Figura 19**

Captura de pantalla panel de administración con detalle de API Gateway de AWS



Nota. Adaptado de Consola de administración de AWS. por Amazon Web Services, (sf).

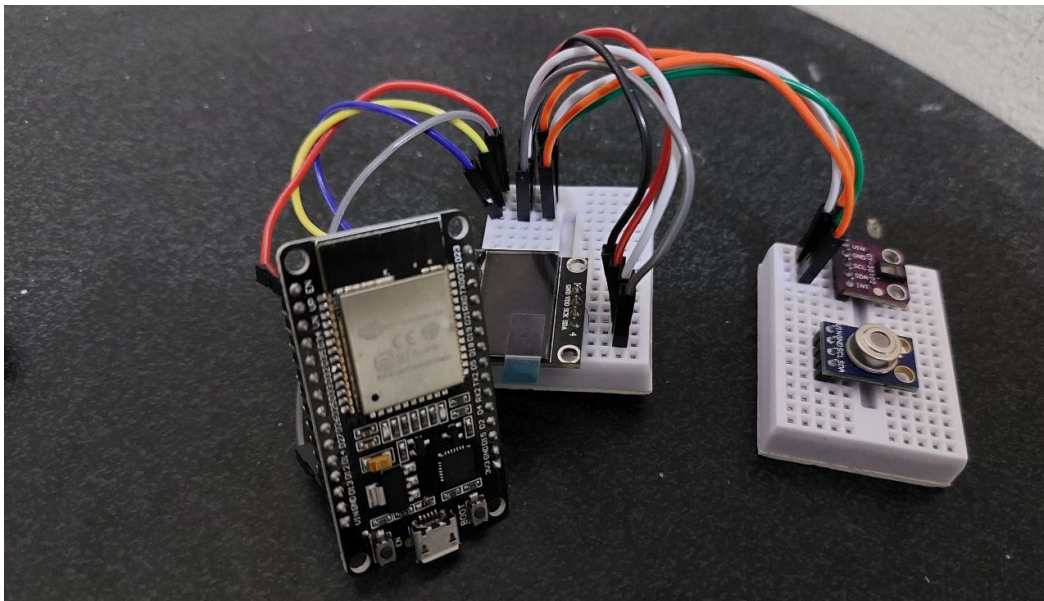
**5.5.2.4 Selección y pruebas iniciales de la tecnología front-end.** Para la tecnología front-end se utilizó React, para la web React.js y para la aplicación móvil React Native, estas dos en Javascript. Estas librerías aportan robustez al proyecto, además de tener varias librerías compatibles adicionales que permiten realizar implementaciones necesarias para llevar a cabo el cumplimiento de requerimientos, adicionalmente existe un conocimiento previo por parte de los miembros del equipo.

Con el fin de realizar la exploración de estas tecnologías se crearon proyectos iniciales, se probaron algunas librerías y componentes, y una mirada inicial a la integración con AWS.

**5.5.2.5 Montaje inicial de componentes hardware en placa de pruebas.** Un montaje inicial se realizó en la protoboard (ver figura 20) con el fin de verificar las conexiones adecuadas entre el microcontrolador y los sensores; la distribución de estos componentes no fué ciertamente evaluada para trabajar en condiciones ideales con bajos niveles de ruido, simplemente considerando que cumpliera los aspectos necesarios para funcionar con la toma de datos correcta y tuviese el paso de energía estipulado en las hojas técnicas, para no generar inconsistencias o variaciones inesperadas en la lectura de datos. La fuente de energía utilizada es un cargador de smartphone con micro usb, la cual funcionó sin problemas durante las pruebas realizadas.

### Figura 20

Montaje en protoboard de prototipo dispositivo hardware

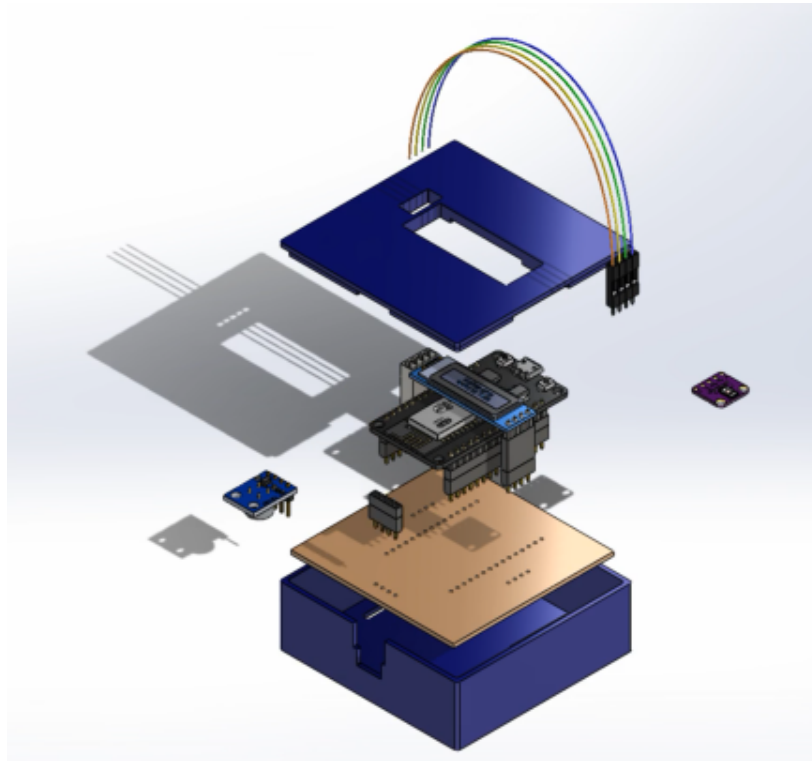


Se evidenció que el funcionamiento de dicho sistema fue exitoso, así que se procedió a replicar las conexiones esta vez ajustadas a la placa diseñada específicamente para el montaje; Con la ayuda de una herramienta de soldadura, se ensamblaron las partes dentro la placa del circuito, y se acomodaron de manera que se logre dejar una zona libre y apta para la recepción de datos cuando el usuario introduzca el índice derecho (o izquierdo), proporcionando una medición alejada en gran parte de los rayos de luz externos, otros cuerpos cercanos y demás factores que puedan alterar significativamente la medición. Para esta implementación también se hace uso del cargador de smartphone como fuente de energía, ya que cumple con proporcionar un voltaje estable a lo largo del periodo de uso del dispositivo.

A continuación, se presentan las partes del dispositivo hardware en un simulador 3D y luego el resultado final después del ensamblaje.

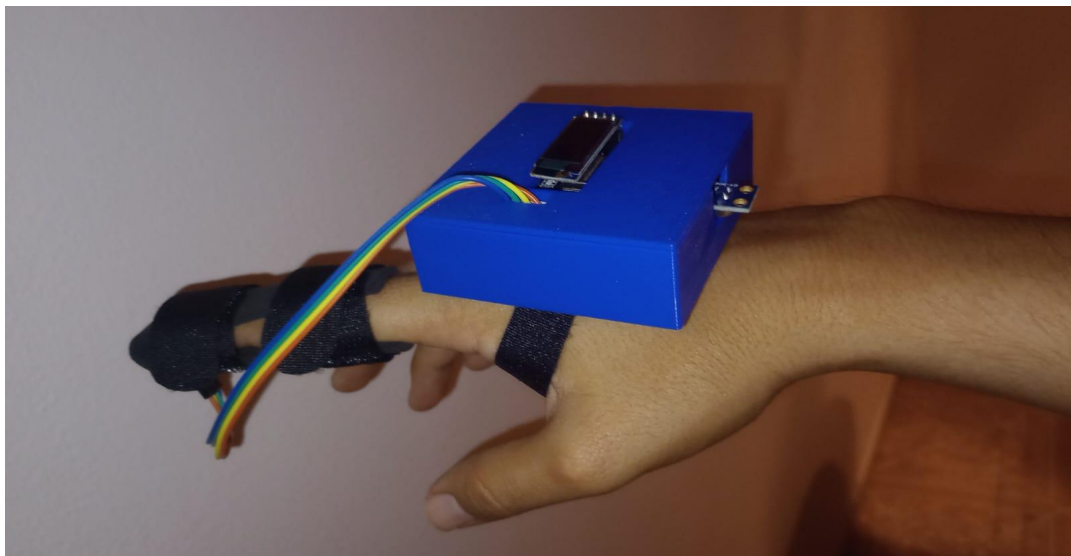
### **Figura 21**

Modelo 3D final del dispositivo hardware sin ensamblar.



**Figura 22**

Montaje final dispositivo hardware terminado y ajustado en la superficie de la mano

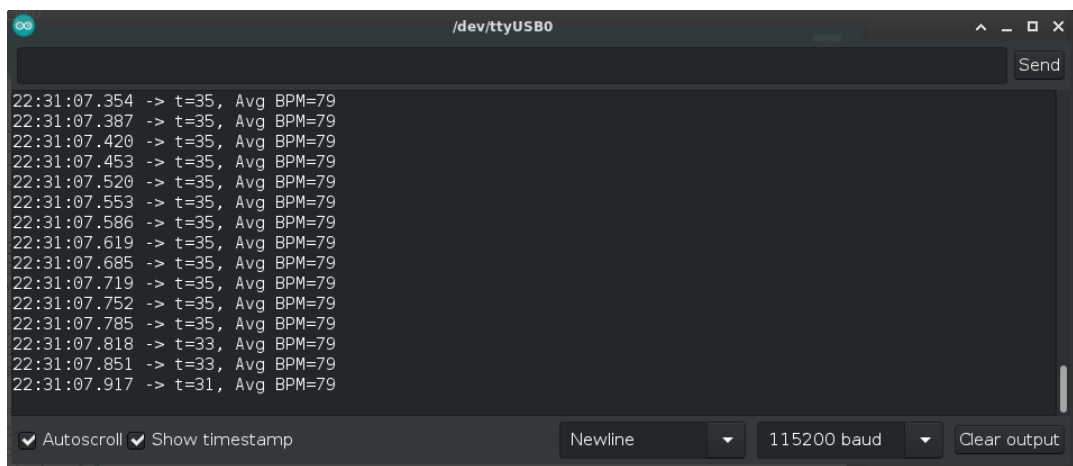


**5.5.2.6 Desarrollo y pruebas de adquisición de datos.** Se realizó el desarrollo para la placa de desarrollo ESP32 Devkit en el cuál permitiera adquirir los datos de las variables definidas

Mediante el monitor serial del dispositivo se puede ver cómo se imprimen las variables adquiridas

### Figura 23

Consola de monitor serial con impresión de las variables capturadas por el dispositivo



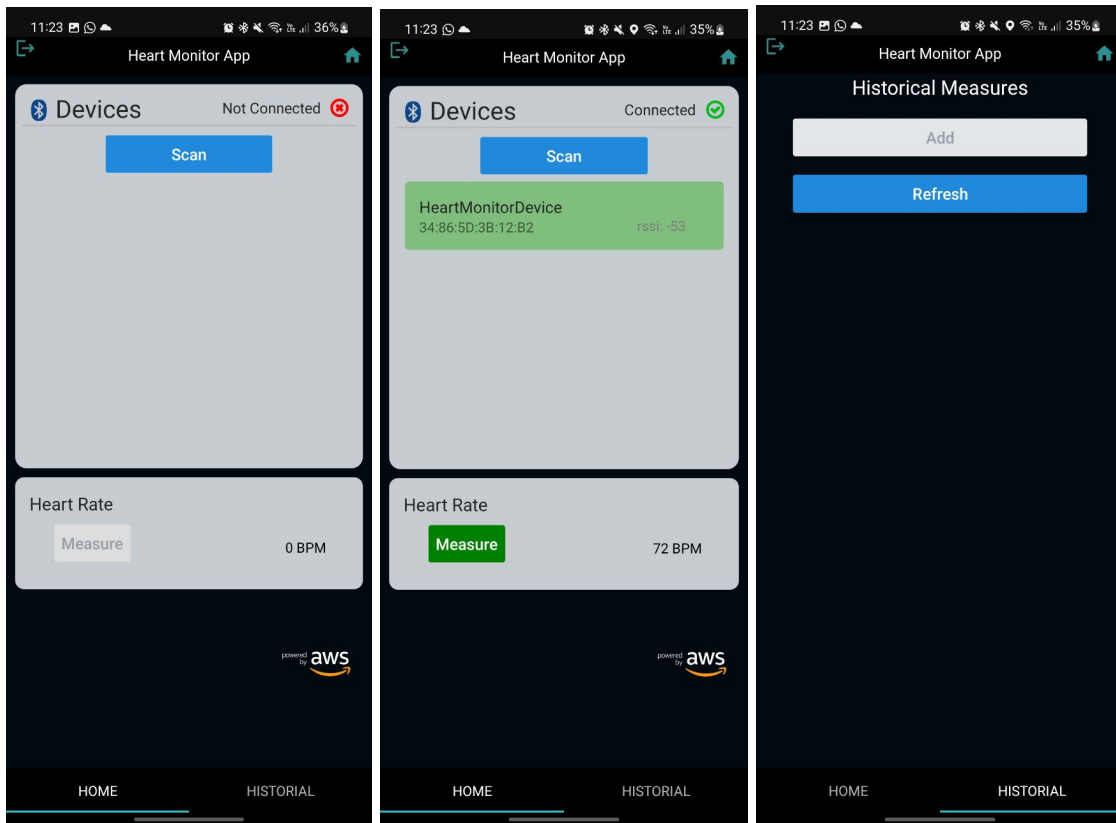
```
/dev/ttyUSB0
22:31:07.354 -> t=35, Avg BPM=79
22:31:07.387 -> t=35, Avg BPM=79
22:31:07.420 -> t=35, Avg BPM=79
22:31:07.453 -> t=35, Avg BPM=79
22:31:07.520 -> t=35, Avg BPM=79
22:31:07.553 -> t=35, Avg BPM=79
22:31:07.586 -> t=35, Avg BPM=79
22:31:07.619 -> t=35, Avg BPM=79
22:31:07.685 -> t=35, Avg BPM=79
22:31:07.719 -> t=35, Avg BPM=79
22:31:07.752 -> t=35, Avg BPM=79
22:31:07.785 -> t=35, Avg BPM=79
22:31:07.818 -> t=33, Avg BPM=79
22:31:07.851 -> t=33, Avg BPM=79
22:31:07.917 -> t=31, Avg BPM=79
```

**5.5.2.7 Diseño y desarrollo front-end.** Ya que se cuenta con dos proyectos frontend que serán parte de la plataforma, se dividió el desarrollo en estos, la aplicación móvil y la plataforma web.

**5.5.2.7.1 Aplicación móvil.** Se definieron algunas pantallas iniciales, para la aplicación móvil, que fueron desarrolladas en React Native, como previamente se definió.

### Figura 24

Pantallas diseñadas para funcionamiento de la aplicación móvil

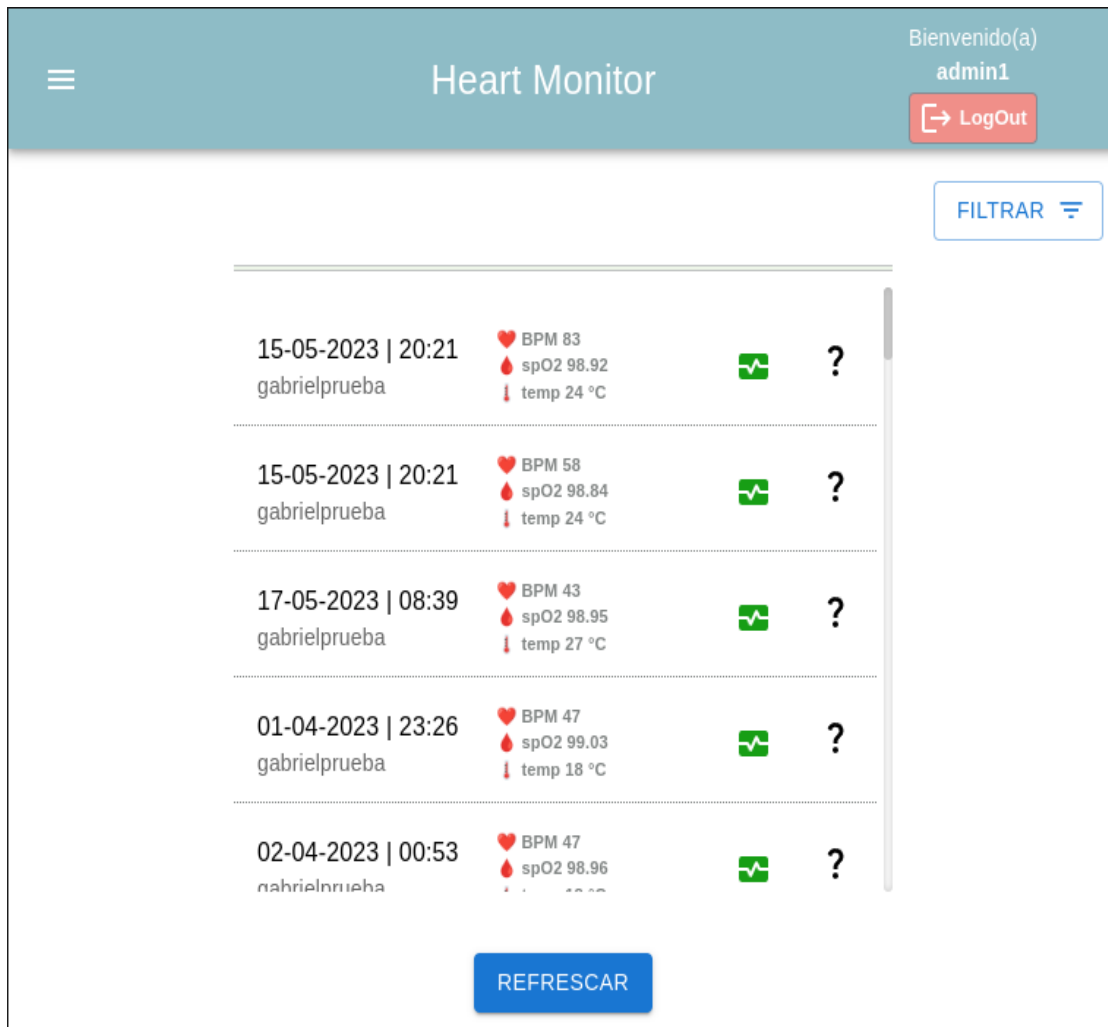


Inicialmente se trata de pantallas sin funcionalidades, siendo sólo visuales sin implementar conexiones o demás lógica.

**5.5.2.7.2 Plataforma web.** Para la plataforma web se definió que la pantalla principal fuese el listado de las aplicaciones y un pantalla secundaria accesible por medio del menú superior la cuál muestra el listado de usuarios editable.

### Figura 25

Pantalla principal de plataforma web que muestra listado de mediciones



**Figura 26**

Pantalla de usuarios de plataforma web

Nombre de usuario *	Max heart...	Min heart ...	Min SpO2	Habilitado	
<input type="text" value="gabrielprueba"/>	<input type="text" value="80"/>	<input type="text" value="50"/>	<input type="text" value="89"/>	<input checked="" type="checkbox"/>	<input type="button" value="GUARDAR"/>
<input type="text" value="miguelprueba"/>	<input type="text" value="120"/>	<input type="text" value="59"/>	<input type="text" value="90"/>	<input checked="" type="checkbox"/>	<input type="button" value="GUARDAR"/>

**5.5.2.8 Exploración e integración de servicios en la nube con front-end.** Dentro de la exploración inicial de los servicios de AWS, se evidenció la compatibilidad con diferentes librerías y frameworks frontend, para esto se usa una librería propia de Amplify que permite implementar dentro de React los diferentes componentes de backend, y así compatibilizar los componentes front y back.

### Figura 27

Comandos utilizados para la implementación de AWS Amplify en React

```
gavo@debian-HP-laptop:/media/gavo/Local/Proyectos/Dev/React Native/heartMonitor$  
npm install -g @aws-amplify/cli  
gavo@debian-HP-laptop:/media/gavo/Local/Proyectos/Dev/React Native/heartMonitor$  
amplify init  
gavo@debian-HP-laptop:/media/gavo/Local/Proyectos/Dev/React Native/heartMonitor$  
npm install aws-amplify @aws-amplify/ui-react  
Amplify in the client is to install the necessary dependencies: Set up fullst
```

Una vez instaladas las librerías de amplify se realizan implementaciones dentro del código que permiten hacer llamados a los servicios de AWS

Los servicios de AWS, específicamente Amplify ofrecen certificados SSL que permiten que la información viaje cifrada a través de la red, bajo el protocolo HTTPS, garantizando así que no estos datos no puedan ser accedidos en claro.

### Figura 28

Bloques de código para la implementación de AWS Amplify en React

```
//AWS  
import Auth from '@aws-amplify/auth';  
import API from '@aws-amplify/api';  
import {withAuthenticator} from 'aws-amplify-react-native';
```

```
API.put(apiName, path, myInit)  
  .then(response => {  
    console.log(response);  
  })  
  .catch(error => {  
    console.log(error.response);  
  });
```

**5.5.2.9 Definición y desarrollo de modo de conexión entre el dispositivo hardware y la aplicación móvil para transmisión de datos.** La tarjeta ESP32 nos ofrece la posibilidad de realizar conexiones a WIFI y también a Bluetooth; las dos opciones fueron exploradas,

evaluando las características que podrían ofrecer para el funcionamiento de la plataforma. La conexión WIFI a pesar de brindar ciertas facilidades, cómo la opción de comunicación TCP o la conexión directa con los servicios en la nube, a través de internet, impone la necesidad de tener una red WIFI disponible en todo momento, y adicionalmente implica un consumo de energía más alto frente a otras opciones.

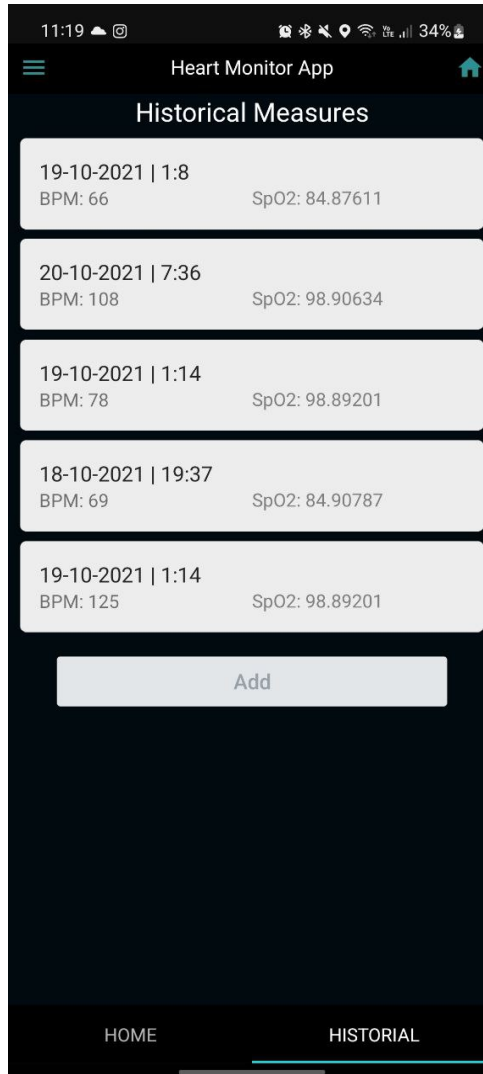
La conexión bluetooth, por el contrario, requiere solamente tener el dispositivo móvil, el cuál puede ser usado cómo puente, ya sea vía WIFI o datos móviles, para la conexión con la nube, además bluetooth cuenta con una implementación llamada BLE o bluetooth de baja energía, que permite tener un menor consumo de energía a la hora de funcionar, optimizando así el uso del dispositivo. Adicionalmente cuenta, por ejemplo con comunicación asíncrona sin conexión, lo que permite una comunicación en caso de que la conexión se haya perdido temporalmente.

**5.5.2.10 Integración completa entre dispositivo hardware, aplicación móvil, servicios en la nube y web.** Habiendo definido y probado previamente el modo de conexión entre el dispositivo IoT con la aplicación móvil y de el componente front-end con el ambiente en la nube, integrar estos 2 componentes resultó en enviar a través de internet a la nube los datos recibidos desde el dispositivo de hardware a la aplicación móvil.

Dentro de la aplicación móvil se creó un apartado de historial de mediciones, dónde se pueden visualizar las mediciones hechas previamente y agregar nuevas, estando estas sincronizadas con la persistencia en la nube.

**Figura 29**

Pantalla de aplicación móvil con historial de mediciones

**Figura 30**

Captura de pantalla panel de administración de DynamoDB mostrando la tabla de registros de mediciones

The screenshot shows the 'NearData-Dev' interface. At the top, there's a 'Scan/Query items' section with a sub-header 'Expand to query or scan items.' Below this, a status bar indicates 'Completed' with 'Read capacity units consumed: 1'. The main section is titled 'Items returned (5)' and includes a refresh button, an 'Actions' dropdown, and a 'Create item' button. Below the controls is a table with 5 rows and 5 columns: 'id', 'data', 'date', and 'userName'. Each row contains a checkbox, a truncated ID, a truncated data object, a date, and the username 'gabrielprueba'.

<input type="checkbox"/>	id	data	date	userName
<input type="checkbox"/>	gabrielprueba2021-1...	{"data":{10...	19-10-202...	gabrielprueba
<input type="checkbox"/>	gabrielprueba2021-1...	{"data":{11...	20-10-202...	gabrielprueba
<input type="checkbox"/>	gabrielprueba2021-1...	{"data":{10...	19-10-202...	gabrielprueba
<input type="checkbox"/>	gabrielprueba2021-1...	{"data":{11...	18-10-202...	gabrielprueba
<input type="checkbox"/>	gabrielprueba2021-1...	{"data":{10...	19-10-202...	gabrielprueba

## 5.6 Evaluación y pruebas del prototipo

Se evidencia por parte de los participantes del proyecto que el prototipo es netamente funcional y está listo para ponerse a prueba con otros voluntarios y recibir las críticas desde su perspectiva para los diferentes factores que influyen en el prototipo, como lo son su estructura, diseño, hardware, software, fiabilidad de los datos y experiencia de usuario.

Para evaluar estos factores y abarcar de manera amplia las posibles críticas y sugerencias, se planteó el formato adjunto en los anexos A, B, C y D con el fin de proporcionar evidencia de la toma de datos a una población de muestra.

La población de muestra a quien va dirigida la demostración de este prototipo es un selecto grupo estudiantil de la facultad de salud UIS, proporcionado gracias a la Dra. Laura Valencia, los cuales cuentan con experiencia principalmente en el área de la salud.

### 5.6.1 Realización de pruebas, toma de datos

Partiendo de la puesta en común para los participantes de esta exposición, se fue realizando gradualmente la presentación hasta llegar al punto de recolectar las pertinentes críticas de los estudiantes y docente encargada (Dra. Laura Valencia); la ruta que se siguió para la presentación fue la descrita en las siguientes subsecciones de este apartado.

**5.6.1.1 Presentación de funcionalidades del prototipo.** A través de la justificación del proyecto, se va haciendo una introducción con la razón por la que nació el proyecto, explicando qué propósito tiene el prototipo y para qué público va dirigido en específico, y siendo así, se formó el contexto para el cual se aplicaría dicha tecnología.

Se indica brevemente cómo está diseñado y estructurado el dispositivo, qué factores intervienen, cuáles son los mecanismos usados y cuál debería ser su correcto funcionamiento. También, se indica el flujo que se debe seguir en el sistema para obtener como resultado final la correcta medición de los signos vitales junto con su representación gráfica y signos de alerta (sólo en caso de superar los límites configurados en la plataforma).

Se dió a conocer cómo configurar el aplicativo móvil y configurar los parámetros de la plataforma web, así como también se explicó cómo funcionan las señales de alerta y el significado que tienen para el usuario y/o administrador.

Se realizó un flujo de prueba para demostrar la conexión bluetooth del aplicativo móvil al dispositivo y una vez enlazado, se configuraron los parámetros de alerta para el paciente en la plataforma, se hizo la toma de los datos y se visualizó el histórico tanto del aplicativo, como de la plataforma web; también se indicó cómo interpretar los signos de alerta.

### **Figura 31**

Fotografía de miembros del proyecto en presentación ante estudiantes de medicina



**Figura 32**

Fotografía de miembros del proyecto junto con estudiantes de medicina



**5.6.1.2 Interacción del público con el prototipo.** Los voluntarios se acercaron en grupo para interactuar con el prototipo y asimismo, uno de ellos, contribuyó para realizar las pruebas necesarias, tomando los datos arrojados por el dispositivo en la plataforma, y así se llevó a cabo una demostración generalizada con el flujo propuesto con el prototipo. Luego, se procedió a

tomar los datos de los voluntarios uno a uno con la medición alterna del dispositivo médico para poder comparar qué tanto divergen entre sí.

Durante la demostración hubo preguntas puntuales del público, críticas y sugerencias, las cuales se pueden observar en la guía de pruebas (ver Anexo C). Las preguntas fueron contestadas y las sugerencias tomadas con su debido reconocimiento.

### Figura 33

Fotografía de miembros del proyecto en presentación ante estudiantes de medicina



**5.6.1.3 Interacción con dispositivos y simuladores del laboratorio clínico.** Durante la visita a la facultad de salud, se obtuvo el acceso para el laboratorio de simulación de parto y allí se interactuó con un maniquí que contaba con diversas funciones para simular situaciones que requieren atención médica; lo cual permitió probar el prototipo en una de las manos de la maniquí. Se hizo el montaje y se captaron las señales que se estaban induciendo controladamente

mediante un equipo de cómputo a la maniquí simulador. Se captó el pulso y el oxígeno en sangre. En esta práctica principalmente se pudo corroborar la captura de señales y estabilidad que se presenta en comparación con el prototipo y el simulador, así como también la respuesta a variaciones de signos repentinamente.

**Figura 34.**

Toma de pulso a modelo de persona en simulador de parto.



**Figura 35.**

Dispositivos para simulación en sala de parto.



**5.6.1.3 Toma de datos de voluntarios.** Uno a uno los voluntarios fueron pasando para realizar la toma de datos de manera alternativa entre el prototipo de monitoreo remoto de pacientes y los dispositivos profesionales de uso médico los cuales eran el pulsioxímetro, que mide el oxígeno en sangre y el pulso cardíaco (ubicado en un dedo de la misma mano en el que se ubicaba el prototipo), asimismo con un termómetro para cuando se medía la temperatura del paciente (este se ubicaba en la axila del paciente).

### **Figura 36**

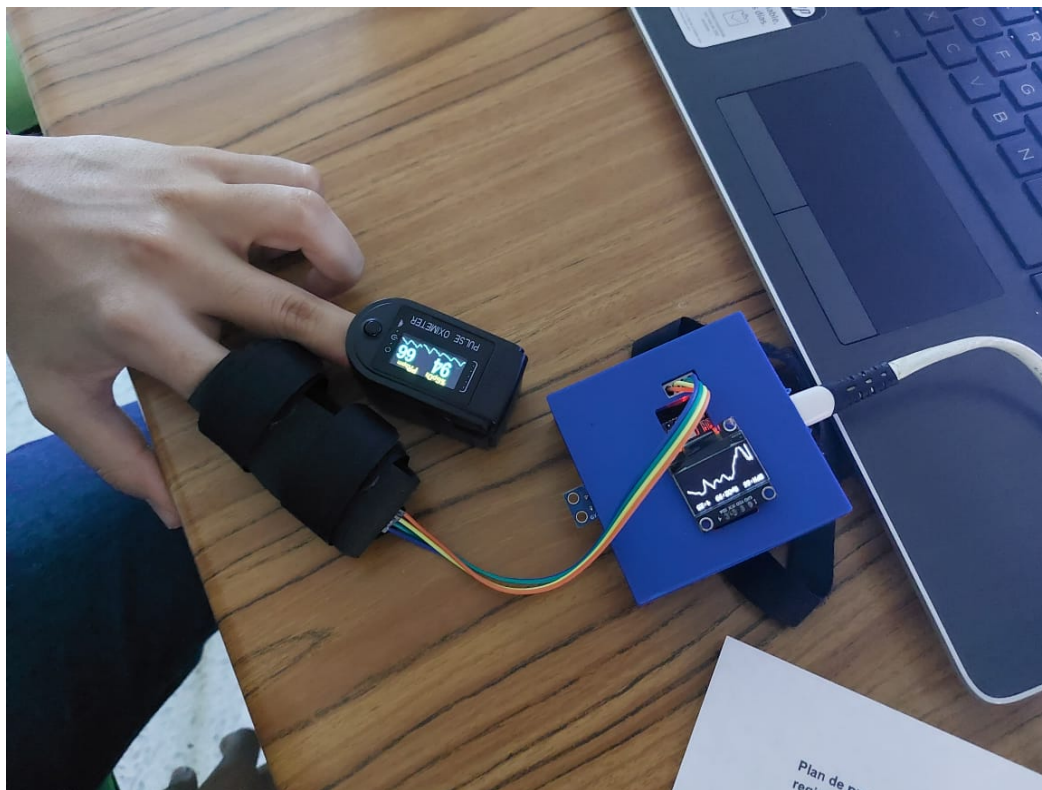
Fotografía voluntario en toma de mediciones



Se tomaron los datos exactamente en el mismo momento una vez estaban estabilizados ambos dispositivos, con el fin de asegurar tener las condiciones similares para ambos dispositivos; siendo así, se tomó los datos de una muestra de 5 voluntarios y se obtuvieron los resultados que se encuentran presentados posteriormente (ver Tabla 3 en la sección 5.4.1.3) y en el plan de pruebas (ver Anexo A).

### **Figura 37**

Fotografía prototipo y pulsioxímetro comercial realizando mediciones en paralelo



Es de resaltar que las condiciones de salud de estos voluntarios era normal, así que la razón principal de la toma de datos era evidenciar qué tan semejantes son los datos arrojados por ambos dispositivos. La toma de los datos se llevó a cabo con éxito, reportándose en las hojas guía para las pruebas y visualizando en el histórico de la plataforma web, así como en la interfaz visual de los dispositivos médicos.

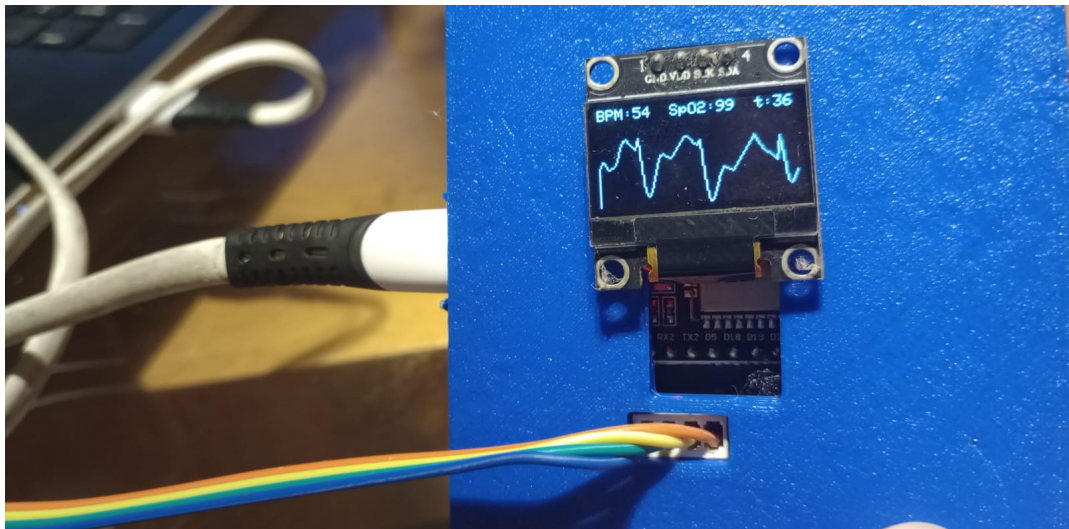
### **Figura 38**

Fotografía co-tutora y miembro del proyecto, en toma de datos con prototipo y pulsioxímetro comercial



**Figura 39**

Fotografía de toma de temperatura por parte del sensor del prototipo



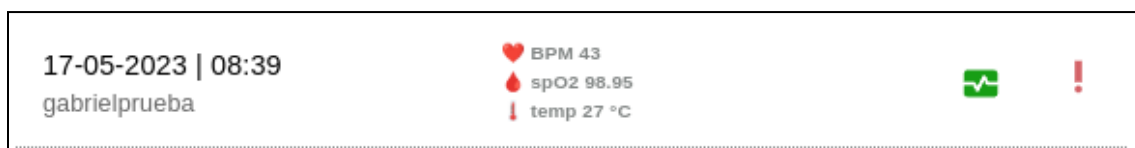
**5.6.1.4 Pruebas mediciones registradas en web.** Se realizaron mediciones automáticas y ejecutadas manualmente con el fin de verlas registradas en la página web.

Las mediciones automáticas ejecutadas por las alertas lanzadas con valores de las mediciones fuera de rango también pudieron ser probadas, viendo la alerta reflejada en la aplicación y también la medición en la web.

Cabe recalcar que estas pruebas se hicieron bajo condiciones favorables, cómo una buena conexión a internet, un paciente que permaneciera sin hacer movimientos bruscos, además de una fuente de alimentación.

### Figura 40

Medición en la plataforma web heart monitor, tomada en ejecución de las pruebas tomada debido a las alertas



*Nota.* Se visualizó en la pantalla del dispositivo el valor de BPM, así cómo la alerta mostrada en la aplicación.

### Figura 41

Lista de mediciones registradas en la plataforma web heart monitor

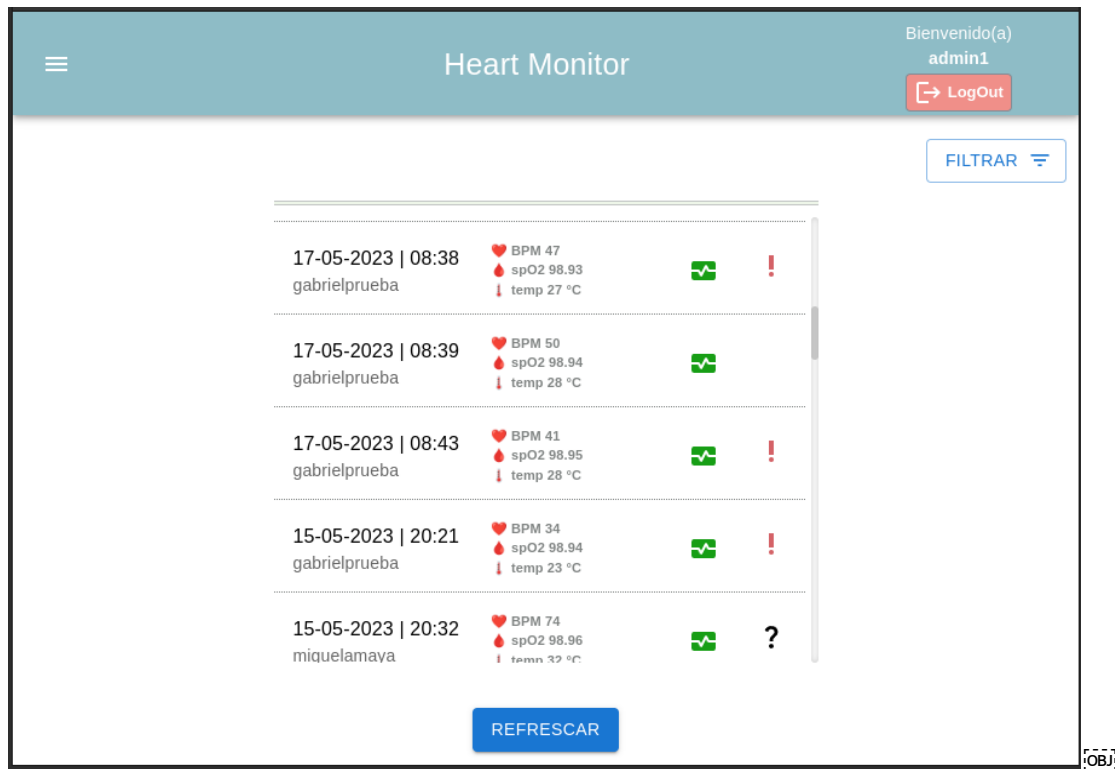
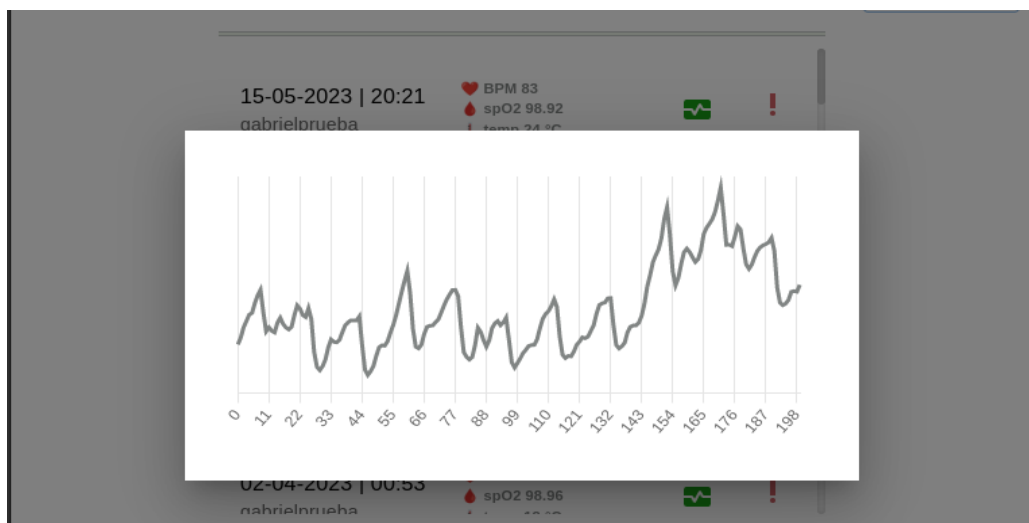


Figura 42

Visualización del gráfico cardiaco de una medición de en la plataforma web

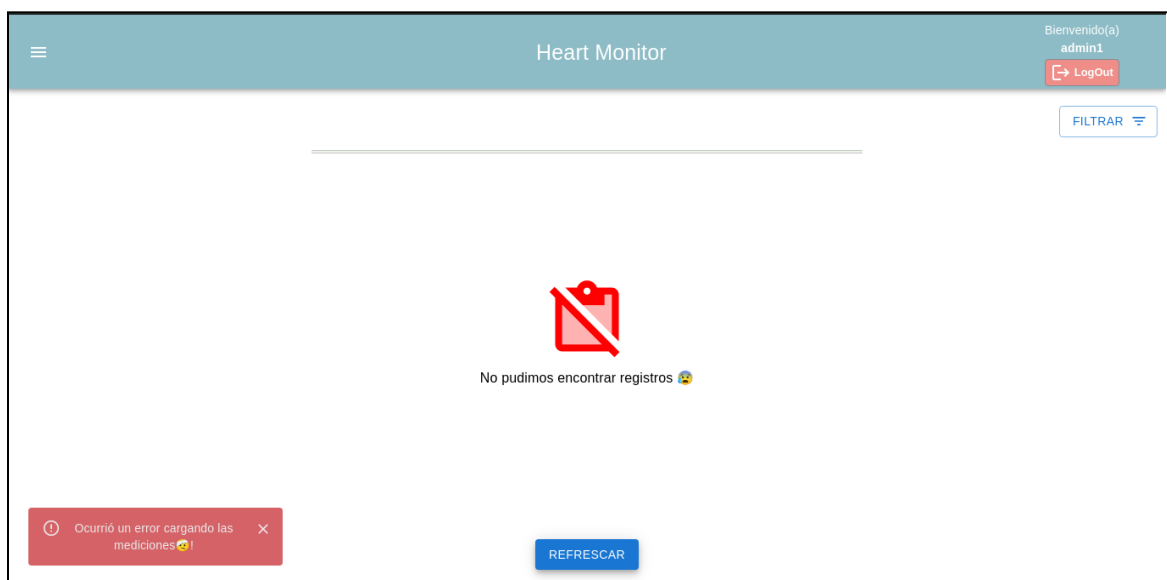


**5.6.1.5 Pruebas de errores en la página web.** Se realizaron pruebas de los posibles errores que pueden ocurrir en la página web y cómo se da el manejo de estos; por lo tanto, se simularon errores procedentes de problemas de conexión, desde el backend, entre otras cosas.

Inicialmente, se probó un error en la carga de registros.

### Figura 43

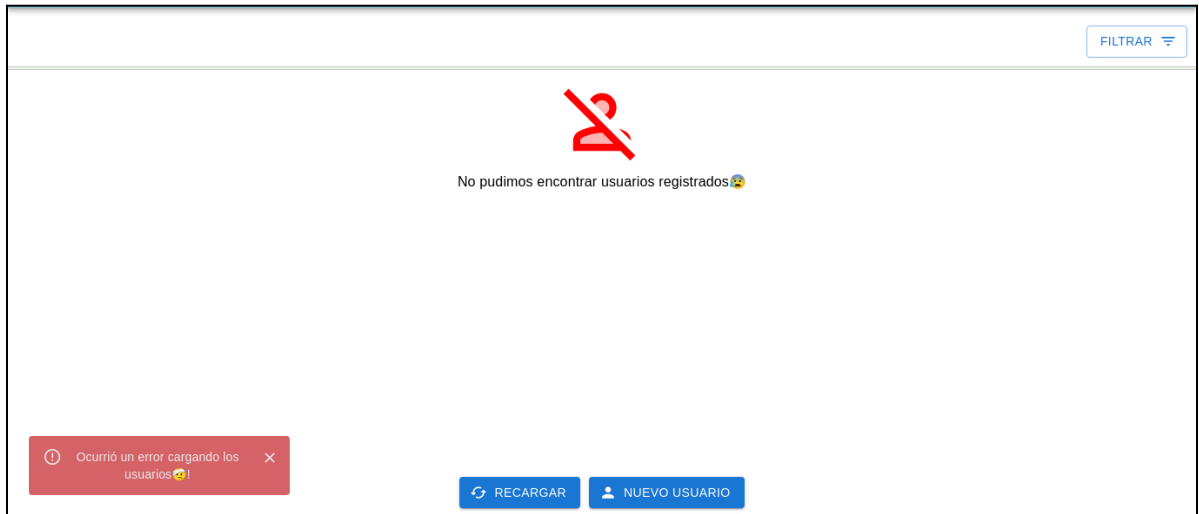
Carga de registros cardiacos con un error



Se apagó el API de usuarios y se observó cómo se vería reflejado en la página web.

### Figura 44

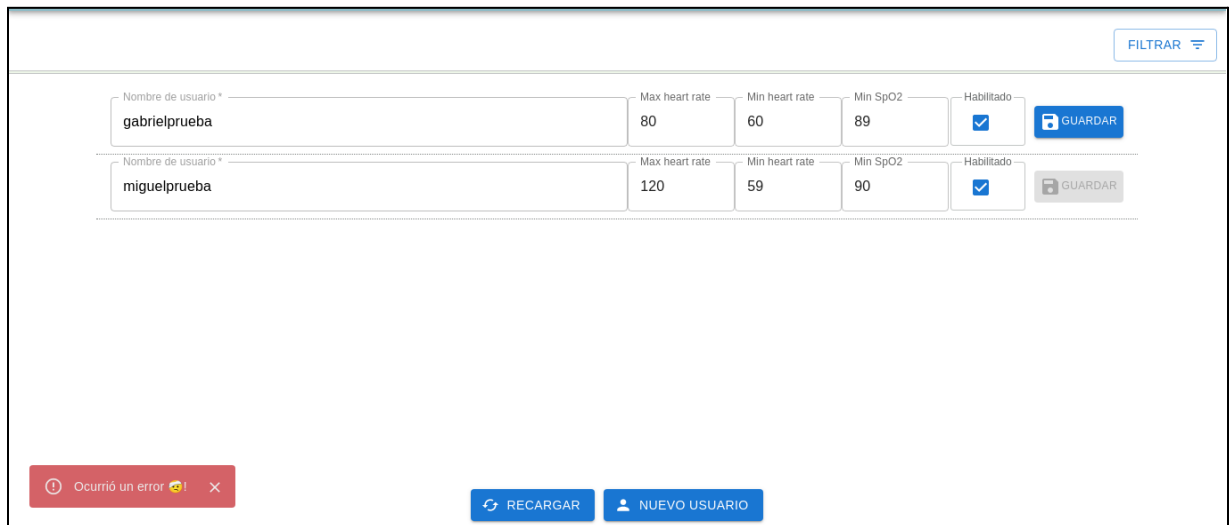
Carga de listado de usuarios con un error



De igual forma se pudo probar qué ocurriría si se presentara un error guardando los usuarios.

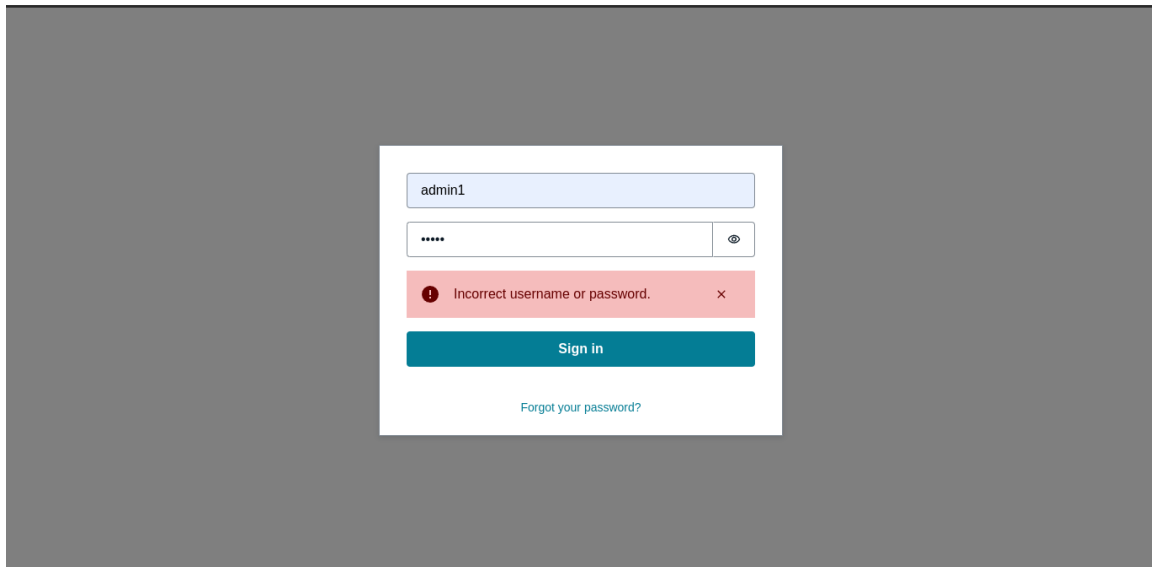
**Figura 45**

Guardado de modificación de usuarios con un error



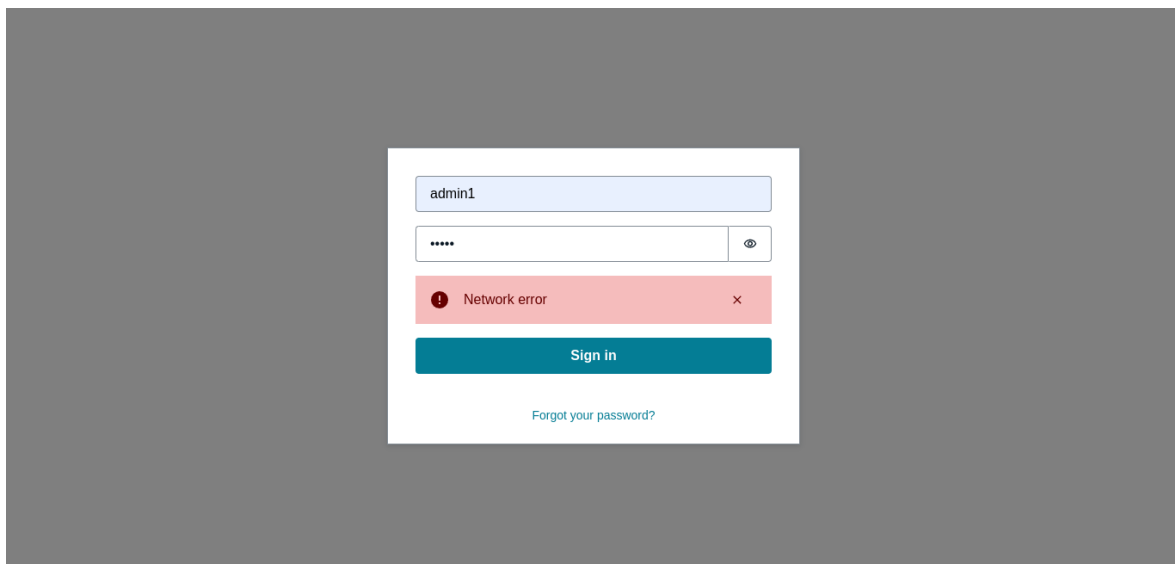
**Figura 46**

Error por contraseña incorrecta login



**Figura 47**

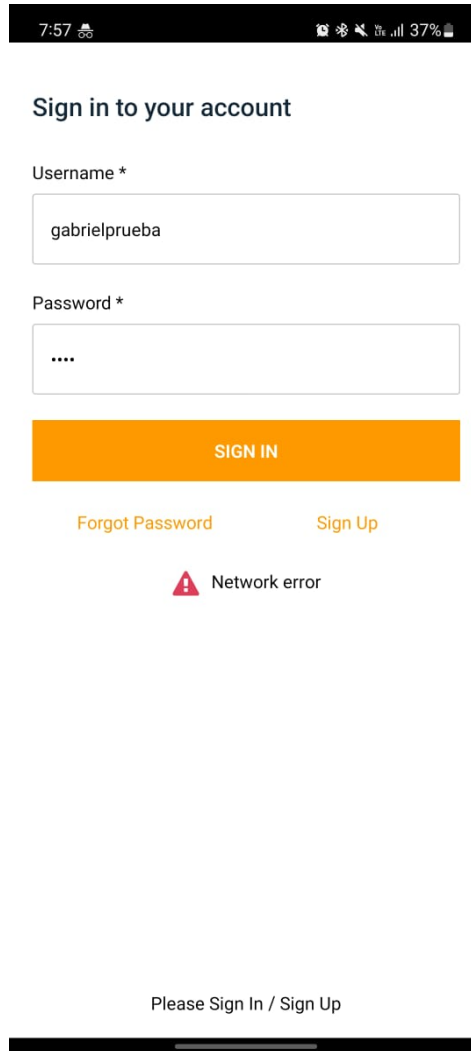
Error por problemas de red en el login



**5.6.1.6 Pruebas de errores en la aplicación móvil.** Entendiendo que pueden ocurrir errores ocasionados por las condiciones en las que funciona la app, previendo problemas de conexión, o entradas no válidas se realizaron pruebas de estos escenarios para ver el manejo de errores dado.

**Figura 48**

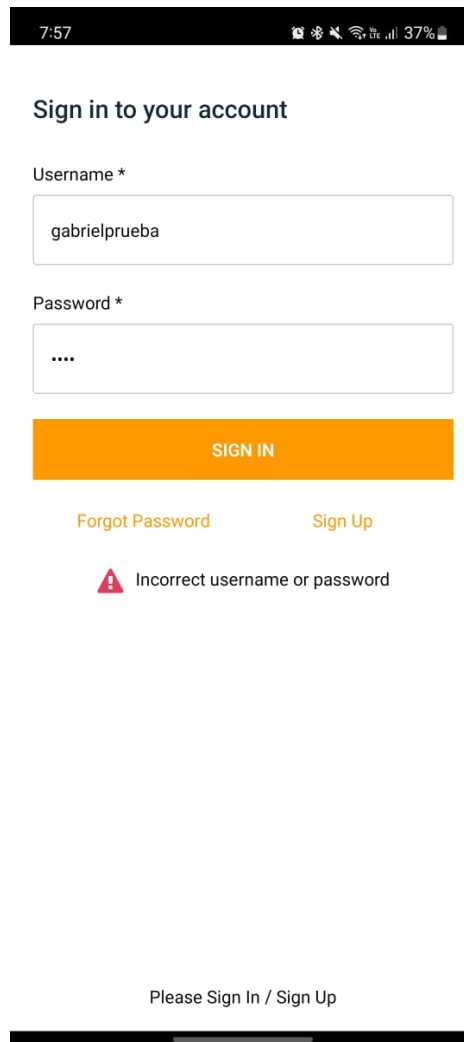
Error por problemas de red en el login en la aplicación móvil



The screenshot shows a mobile application interface for logging in. At the top, a status bar displays the time 7:57, signal strength, Wi-Fi, and 37% battery. The main heading is "Sign in to your account". Below it, there are two input fields: "Username \*" containing "gabrielprueba" and "Password \*" containing four dots. A prominent orange "SIGN IN" button is centered below the fields. Underneath the button are two links: "Forgot Password" and "Sign Up". A red warning icon with a triangle and exclamation mark is followed by the text "Network error". At the bottom of the screen, a footer message reads "Please Sign In / Sign Up".

**Figura 49**

Error generado por ingreso erróneo de las credenciales en el login



**5.6.1.7 Comparación entre dispositivos.**

**Tabla 3.**

Comparativa de medición de pulsos por minuto [BPM] tomados por el pulsioxímetro de uso médico y por el sensor de pulsioximetría del prototipo.

Toma	Dispositivo[bpm]	Prototipo [bpm]
1	74	77
2	66	67

3	66	69
4	59	60
5	64	63
6	71	70

**Tabla 4.**

Comparativa de medición de oxígeno en sangre [spO2] tomados por pulsioxímetro de uso médico y por el sensor de pulsioximetría del prototipo.

Toma	Dispositivo[spO2]	Prototipo [spO2]
1	95	99
2	94	99
3	95	99
4	98	99
5	97	99

**Tabla 5.**

Comparativa de medición de grados celsius [°C] tomados por termómetro de uso médico y por el sensor del prototipo

Toma	Dispositivo[°C]	Prototipo [°C]
1	36.0	36.4
2	36.2	37.0
3	35.7	36.5
4	35.9	36
5	36.1	35.7

### ***5.6.2 Recepción de críticas y sugerencias***

Algunas de las principales observaciones giraron en torno al diseño del prototipo, puesto que se sugería que el mismo tuviese un tamaño menor, así como un diseño que permitiese una mayor movilidad por parte del usuario.

Por otro lado, se hicieron sugerencias de mejoras en cuanto al tratamiento de datos y personalización de los filtros, por ejemplo, poder clasificar la gravedad de las alertas, o que los datos del paciente estuvieran enlazados a los sistemas de información hospitalarios con el fin de brindar mayor información sobre el estado del paciente.

**5.6.2.1 Críticas del público general (estudiantes).** Habiendo recolectado una serie de críticas evidenciadas en el plan de pruebas (ver Anexo C) se puede concluir que, desde la perspectiva del público y a nivel de diseño, puede ser más cómodo si la ubicación del dispositivo es en la muñeca mas no en el dorso de la mano; asimismo, la ubicación del sensor de temperatura podría estar en la misma ubicación del sensor de pulsioximetría, es decir, el dedo índice, ya que esto no influiría considerablemente en la toma de los datos y brindaría mayor comodidad al paciente.

Otra sugerencia importante fué que se implementara una batería portátil para el dispositivo y siendo así no tuviese que estar conectado a una fuente directa y su funcionamiento no dependa de una conexión con cable, ya que esto podría impedir el desplazamiento de los pacientes que tuviesen que ser monitoreados muy frecuentemente.

Con el fin de mejorar la precisión al momento de tomar los datos, los participantes hicieron la recomendación de que la férula debería cubrir el mayor espacio posible en el lugar donde se realiza la medición, ya que la luz interviene en este sensor y puede causar ruido en la

recolección de los datos; del mismo modo, la ubicación del sensor de pulsioximetría debería fijarse en la férula en posición frente a la yema del dedo mas no en el extremo del dedo.

También, se cuestionó acerca del cable que conecta los sensores del dedo índice al microcontrolador, ya que este podría parecer corto para varios pacientes, por lo tanto, uno de los participantes sugirió que este cable pudiese ser más flexible y extenso.

Acercas del aplicativo móvil, los voluntarios compartieron su conformidad con el diseño y funcionalidades con las que cuenta este; sólo hubo una intervención acerca de la presentación de los datos, y es que para muchos casos es innecesario la representación gráfica del monitoreo cardiaco, ya que esta app estaría siendo utilizada por inexpertos, o en otras palabras, los encargados del paciente; Estos datos sólo podrían ser interpretados correctamente por el administrador, es decir, quien está a cargo de la plataforma web (médico o personal profesional del sector salud).

Para prevención ante situaciones de riesgo, se recibió la sugerencia de que se podría mostrar una alerta cuando se pierda la conexión a internet en medio de un monitoreo; y que debería por otros medios informarse tanto al personal encargado como médico sobre la situación.

Finalmente, sobre la plataforma web se recomendó que se creara una prioridad de alertas, es decir, que se pudiese configurar el nivel de riesgo de cada paciente, para dar la atención correspondiente a cada caso por parte de los administradores, que en este caso sería el personal médico y de esta manera, regular la frecuencia con la que se debe monitorear el paciente. También, enfatizaron que sería de gran importancia la vinculación del sistema de información del proveedor de salud, para identificar en conjunto con el historial del paciente, otros factores que pudiesen intervenir en los monitoreos (ver Anexo C).

**5.6.2.2 Críticas del experto (Dra. Laura Valencia).** Se hizo la recomendación de que la férula de dedo debería ser lo suficientemente flexible y dócil como para poderse transportar sin ningún problema para el paciente, esto quiere decir que el movimiento no debería repercutir en la variación de datos que toma el sensor.

Para la importancia que tendría la implementación de este prototipo frente al uso profesional por parte de personal de la salud, la Dra. Laura Valencia puntualizó que se deberían usar sensores que se pudiesen ajustar más a las necesidades que compromete un monitoreo de este nivel, como por ejemplo, usar un sensor de temperatura no infrarrojo sino de contacto, ya que esto aumentaría la precisión y fiabilidad de los datos, así como también aplicaría para el sensor de pulsioximetría.

A lo largo de su orientación, la Dra. Laura Valencia mostró su conformidad con la representación de datos, información, alertas, diseño y funcionalidades de los componentes de la plataforma IOT, resaltando lo interesante e innovador que se muestra el prototipo frente a la situación actual de ambas áreas intervenidas, sistemas y medicina (ver Anexo B).

### ***5.6.3 Interpretación de información y éxito del prototipo***

Culminadas las pruebas, se llegó a la deducción de que el tiempo que tarda el dispositivo en adquirir una cantidad de datos considerable para presentar mediciones precisas, era alrededor de 30 segundos desde el inicio de cada una de las pruebas o cambio de participante. Lo cual también se veía reflejado efectivamente en la representación gráfica de las mediciones cardiacas, tomando un comportamiento de una curva sinusoidal tal como se esperaba.

Se lograron tomar 12 registros exitosos con el dispositivo enlazado a la aplicación móvil y reportados también a la aplicación web. Cada uno de estos representado en los históricos de la

aplicación móvil y página web, con los valores de las mediciones y la representación gráfica de las mismas durante el periodo de la toma.

Se pudo comprobar que los valores representados en la pantalla del dispositivo se corresponden con los capturados mediante la aplicación móvil y posteriormente representados en la página web.

Se pudo observar la alerta en la aplicación móvil y así mismo ver el registro marcado con los signos de alerta definidos en la página web, en el momento en que las mediciones estuvieron fuera de los parámetros establecidos como normales.

La conexión Bluetooth del dispositivo se configuró correctamente cada vez que se intentaba asociar con el aplicativo móvil; este enviaba y recibía datos tal como se había codificado. Tomaba los datos que percibía por medio de los sensores y se representan visualmente en la interfaz gráfica del aplicativo, respondiendo a cada una de las peticiones enviadas desde los botones de la app, para finalmente transmitir esta información a la plataforma web y determinar si estos era un signo de alerta o no. Cada flujo se llevó a cabo sin ningún contratiempo, por lo que a la escala de este prototipo, es de considerarse como un sistema óptimo.

Como proyecto prototipo de plataforma IOT, se evidencia que se realizaron satisfactoriamente las pruebas necesarias para comprobar su funcionalidad en base a los objetivos específicos de esta propuesta. De esta manera, se da por terminada la fase de pruebas.

## **6. Conclusiones**

El proyecto cumple con su objetivo, ya que es totalmente funcional fuera de instalaciones médicas y realiza el monitoreo de los signos vitales, brindando una respuesta óptima si se usa de la manera y en las condiciones adecuadas.

Es indispensable contar con el material adecuado para la implementación hardware y conocer sus límites y funcionalidades a profundidad, ya que esta selección hace que la lectura de datos, capacidades, soporte, entre otras, sea de una determinada comodidad y facilidad para el desarrollo del ambiente software del dispositivo; gracias al conocimiento previo de varios dispositivos electrónicos y capacitación en fichas técnicas de los no familiarizados, se pudo desarrollar ampliamente cada una de las funcionalidades software previstas para el proyecto.

A diferencia de otras metodologías ágiles, la metodología tradicional usada para este proyecto se adecuó de manera que fuese orientada al cumplimiento de objetivos previamente definidos y que durante todo el proceso no serían cambiantes; siendo así, fué más eficiente ya que proponía la consecución de tareas de manera escalonada y donde cada una era esencial en el desarrollo de la posterior. Al final se hacía realimentación y en caso de que fuese necesario otra revisión o corrección, se realizaba siempre y cuando estuviese comprometido el siguiente proceso; Esto brindó la libertad de realizar entregas escalonadas y bien definidas que iban cumpliendo parcialmente los objetivos contemplados, finalizando con la entrega de un prototipo sin necesidad de rediseñar o cambiar sus requerimientos iniciales.

En el IOT, que en el futuro, llenará el mundo de dispositivos conectados a la red para múltiples funciones en la vida diaria, así que, la seguridad no debe tratarse como algo para incluir ligeramente al dispositivo, el enfoque debe ser en todo momento integral y robusto para proteger la privacidad e inhibir el acceso a dicha información importante por parte de delincuentes informáticos; Por eso el mismo nivel de inteligencia que permite a esta solución

realizar sus funcionalidades específicas, debe tener la capacidad de reconocer y contrarrestar las amenazas que pueda enfrentar, por ejemplo, evitar el emparejamiento Bluetooth con un dispositivo no vinculado.

Se construyó una solución que usa controles y realiza monitoreos optimizados para los retos que imponen las aplicaciones que se integran día a día como alternativas basadas en IOT; a partir de esto, se proyecta una reducción de costos en comparación con otras plataformas o dispositivos con propósitos similares, ofreciendo un nivel de precisión similar al de estos, adicionalmente teniendo a favor en este proyecto, el nivel de robustez, accesibilidad y calidad software, junto con la confiabilidad de lectura de datos desde el hardware.

Los usuarios de prueba pudieron experimentar el uso de la plataforma a cabalidad, aseguraron que la plataforma y el dispositivo cumplieran con su propósito principal adecuadamente, resaltando su fácil usabilidad e interpretación de la información generada por el aplicativo. Se conciben sugerencias como el mejoramiento del aspecto visual de la plataforma, tratamiento de datos y mejoras en el diseño del prototipo de hardware.

## **7. Recomendaciones**

Independientemente de las aproximaciones de valores de alerta usados en este proyecto brindados por investigación propia y asesoría de la co-tutora, un valor agregado importante para generar mayor confiabilidad y certeza de la gravedad de la situación sería trabajar con un análisis estadístico o de aprendizaje con IA basándose en conjuntos de datos que brinden información acerca de los valores que indican riesgo o han provocado una respuesta de emergencia en los

pacientes, de manera que se disminuya la cantidad de reportes de alertas en la plataforma, descartando aquellas que representan una variación normal de signos vitales del paciente.

Para el diseño hardware se podrían plantear diversas mejoras desde su aspecto físico para hacerse atractivamente visual, hasta rediseñar la arquitectura para prometer resultados aún más consistentes, es decir, mejoras con respecto a la detección de ruido en los componentes electrónicos y sensores o inclusive en la impresión de la placa de instalación del microcontrolador, ya que con la ayuda de profesionales especializados, se puede revisar con detalle, aspectos como la ubicación de los puntos de conexión en la tarjeta de circuito y así recibir los datos con el mayor aislamiento al ruido exterior posible.

Sería interesante adaptar este prototipo al desarrollo software con apoyo de frameworks especialmente para un futuro desarrollo web, con el fin de facilitar el proceso y de una manera eficiente dar respuesta a las peticiones que se harían desde y hacia el ambiente web, que por otra parte, también se reutilizarían estructuras ya elaboradas que son óptimas y comprobadas para el funcionamiento de los aplicativos web.

La implementación de un historial médico (señales de alerta detectadas en el paciente) en conjunto con una nueva funcionalidad de análisis apoyado en conjuntos de datos, habilitaría la posibilidad de trabajar con inteligencia artificial; de esta manera, se lograría una generación de alertas basada en la evidencia de casos ya transcurridos donde hubo algún padecimiento de otro paciente con anterioridad, esto aumentaría la precisión en la predicción de emergencias.

Aunque en este prototipo no se esté evaluando aún la posibilidad de cobro, uso de credenciales bancarias y datos personales de los pacientes o de los terceros responsables, sería muy importante implementar protocolos de seguridad y tratamiento de datos, ya que, al capturar

información personal del estado médico de un paciente, se está tratando con información que puede estar sometida a regulaciones legales.

Además de los ya implementados es posible generar nuevos filtros y mecanismos que permitan manipular los datos obtenidos y así generar un mayor valor hacía el objetivo de identificar condiciones de los pacientes y clasificarlas con el fin de brindar la atención que se requiera.

### Referencias Bibliográficas

- Aveiga, C. E., Criollo, B. G., & Cruz, A. M. (2018, 30 abril). Monitoreo del ritmo cardíaco a través de dispositivos móviles. *Dominio de las Ciencias*, 4(2).  
<https://dialnet.unirioja.es/servlet/articulo?codigo=6870910>
- Weston, R. (2017, 26 junio). Traditional vs Serverless [Gráfico]. GoCD.  
<https://www.gocd.org/assets/images/blog/serverless-continuous-delivery/traditional-vs-serverless-ee2afc44.jpeg>
- Bayer. (2020, 1 septiembre). Las enfermedades cardiovasculares son la primera causa de muerte en Colombia y el mundo. Bayer-Global.  
<https://www.bayer.com/es/co/las-enfermedades-cardiovasculares-son-la-primera-causa-de-muerte-en-colombia-y-el-mundo>
- Bluetooth SIG. (s. f.). Bluetooth Technology Overview. Bluetooth® Technology Website.  
<https://www.bluetooth.com/learn-about-bluetooth/tech-overview/>.
- Díaz, J. E. (2020, 6 noviembre). Comorbilidades de los fallecidos por COVID-19 según el grupo etario en Colombia. *Repertorio de Medicina y Cirugía*, 29(1).  
<https://revistas.fucsalud.edu.co/index.php/repertorio/article/download/1130/1387/3480>

Díaz-Castrillón, F. J., & Toro-Montoya, A. I. (2020). SARS-CoV-2/COVID-19: el virus, la enfermedad y la pandemia. *Biblioteca Virtual en Salud*, 24(3).  
<https://docs.bvsalud.org/biblioref/2020/05/1096519/covid-19.pdf>

EvaluateMedTech. (2018, septiembre). World Preview 2015, Outlook to 2020. EvaluateMedTech, 1(7).  
<https://info.evaluategroup.com/rs/607-YGS-364/images/WPMT2018.pdf>

González Barajas, J. (2016). Electrocardiograma desde una visión digital [Ilustración]. Bogotá, Colombia: Ediciones USTA. Recuperado de  
<https://bibliotecavirtual.uis.edu.co:4259/es/ereader/uis/68988?page=34>.

Makarian, M. (s. f.). Equivalencia comportamientos del corazón con su representación en un electrocardiograma. [Ilustración]. Melina Makarian.  
[https://www.melinamakarian.com/wp-content/uploads/2010/11/MMHE\\_03\\_021\\_01\\_eps.gif](https://www.melinamakarian.com/wp-content/uploads/2010/11/MMHE_03_021_01_eps.gif)

C.D.Products S.A. (2023, 19 abril). *Oxímetro de pulso o pulsioxímetro CDPC01 - C.D* [Ilustración]. *Products S.A. - CDP. C.D. Products S.A. - CDP.*  
<https://www.cdpsa.eu/producto/oximetro-de-pulso-o-pulsioximetro-cdpc01/>

*Hermosillo Heart Team HHT*. (s. f.). Temperatura Corporal[Ilustración]Hospital CIMA Hermosillo.

<https://www.cimahermosillo.com/es/clinicas-de-especialidad/instituto-de-corazon-cima-hermosillo.html>

Gridling, G., & Weiss, B. (2007). Introduction to Microcontrollers (1.4 ed.). Institute of Computer Engineering - Vienna University of Technology.  
<https://ti.tuwien.ac.at/ecs/teaching/courses/mclu/theory-material/Microcontroller.pdf>

Cloudflare, Inc. (s. f.). *¿Qué es HTTPS?* Cloudflare. Recuperado 26 de noviembre de 2022, de <https://www.cloudflare.com/es-es/learning/ssl/what-is-https/>

*Qué es un certificado SSL: definición y explicación.* (2023, 19 abril). latam.kaspersky.com.  
<https://latam.kaspersky.com/resource-center/definitions/what-is-a-ssl-certificate>

*AWS Management Console.* (s. f.). Amazon Web Services, Inc. <https://aws.amazon.com/console/>

Google Cloud Pricing Calculator. (s. f.). Google Cloud.  
<https://cloud.google.com/products/calculator/>

*Display Pantalla Lcd Oled I2c 0.96 128\*64 7 Pines Arduino.* (s. f.). MercadoLibre. Recuperado 25 de enero de 2023, de [https://articulo.mercadolibre.com.co/MCO-640448813-display-pantalla-lcd-oled-i2c-096-12864-7-pines-arduino-\\_JM](https://articulo.mercadolibre.com.co/MCO-640448813-display-pantalla-lcd-oled-i2c-096-12864-7-pines-arduino-_JM)

*Mlx90614 Sensor De Temperatura Infrarrojo Para Arduino.* (s. f.). MercadoLibre. Recuperado 25 de enero de 2023, de [https://articulo.mercadolibre.com.co/MCO-474543492-mlx90614-sensor-de-temperatura-infrarrojo-para-arduino-\\_JM](https://articulo.mercadolibre.com.co/MCO-474543492-mlx90614-sensor-de-temperatura-infrarrojo-para-arduino-_JM)

*Sensor De Concentración De Oxígeno Max30102.* (s. f.). Cuotas sin interés. Recuperado 25 de enero de 2023, de [https://articulo.mercadolibre.com.co/MCO-634565101-sensor-de-concentracion-de-oxigeno-max30102-\\_JM](https://articulo.mercadolibre.com.co/MCO-634565101-sensor-de-concentracion-de-oxigeno-max30102-_JM)

*Tarjeta De Desarrollo Esp32 Wifi Bluetooth.* (s. f.). Cuotas sin interés. Recuperado 25 de enero de 2023, de [https://articulo.mercadolibre.com.co/MCO-823696589-tarjeta-de-desarrollo-esp32-wifi-bluetooth-\\_JM](https://articulo.mercadolibre.com.co/MCO-823696589-tarjeta-de-desarrollo-esp32-wifi-bluetooth-_JM)

Herrera, K. A. (2020, 4 julio). Incrementa la demanda de oxímetros ante la pandemia de coronavirus. *Noticieros Televisa*. <https://noticieros.televisa.com/ultimas-noticias/incrementa-la-demanda-de-oximetros-ante-la-pandemia-de-coronavirus/>

HUAWEI. (s. f.). HUAWEI Health - HUAWEI Colombia. <https://consumer.huawei.com/co/mobileservices/health/>

Lee, I., & Lee, K. (2015, 1 julio). The Internet of Things (IoT): Applications, investments, and challenges for enterprises. ScienceDirect.

<https://linkinghub.elsevier.com/retrieve/pii/S0007681315000373>

Logicbus S.A. de C.V. (s. f.). Sensores de temperatura de tipo infrarrojos - Logicbus. Logicbus.

<https://www.logicbus.com.mx/infrarrojos.php#:~:text=Los%20sensores%20infrarrojos%20son%20compactos,un%20sorprendente%20tiempo%20de%20respuesta.&text=Realizan%20mediciones%20de%20alto%20rendimiento,temperaturas%20desde%20%2D20%C2%BAC%20a%202000%C2%BAC>.

Novas Peña, D. (2008). Microcontroladores: Arquitectura, programación y aplicación.

ATLANTIC INTERNATIONAL UNIVERSITY Honolulu, Hawaii.

<https://www.aiu.edu/applications/DocumentLibraryManager/upload/Despradel%20Novas%20Pe%C3%B1a.pdf>

Oracle. (s. f.). ¿Qué es el Internet de las cosas (IoT)? Oracle Colombia.

<https://www.oracle.com/co/internet-of-things/what-is-iot/>

Observatorio Nacional de Salud & Instituto Nacional de Salud. (2013, 9 diciembre). Boletín No.

1 [Comunicado de prensa].

<https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/IA/INS/Boletin-tecnico-1-ONS.pdf>

Perry, D. & Wolf, A. (1992). Foundations for the study of software architecture. SIGSOFT Softw. Eng. Notes 17, 4 (Oct. 1992), 40–52. DOI:<https://doi.org/10.1145/141874.141884>

Rose, K., Eldridge, S., & Chapin, L. (2015, octubre). LA INTERNET DE LAS COSAS— UNA BREVE RESEÑA. Internet Society. <https://www.internetsociety.org/wp-content/uploads/2017/09/report-InternetOfThings-20160817-es-1.pdf>

Mell, P., & Grance, T. (2011, septiembre). The NIST Definition of Cloud Computing (NIST Special Publication (SP) 800–145). National Institute of Standards and Technology. <https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf>

IBM Cloud Education. (2022, 10 junio). What is Serverless? IBM. <https://www.ibm.com/cloud/learn/serverless>

Microsoft Azure. (s. f.). ¿Qué es IaaS? Infraestructura como servicio. <https://azure.microsoft.com/es-es/resources/cloud-computing-dictionary/what-is-iaas/>

Microsoft Azure. (s. f.-a). IaaS, PaaS & SaaS [Ilustración]. Cloud Computing Services | Microsoft Azure. <https://azurecomcdn.azureedge.net/cvt-7d12cb554a538170e8ffd3f464d6d95301d16aeecba8d10775a17b0cdc5f7a2/images/page/resources/cloud-computing-dictionary/what-is-iaas/iaas-paas-saas.png>

Rodríguez, J. & Argüello E. (2020). Diseño y construcción de un sistema basado en IoT para supervisión remota de la frecuencia cardíaca y el tono simpático a partir de la señal fotopletimográfica (PPG). [Tesis para optar el título de ingeniero electrónico]. Universidad Santiago de Cali.

Rodríguez-Gómez, R. (2019, 31 agosto). Internet de las cosas: Futuro y desafío para la epidemiología y la salud pública | Universidad y Salud. Universidad y Salud. <https://revistas.udenar.edu.co/index.php/usalud/article/view/3628>

Triana, F. J. F. (2020, 1 mayo). COVID-19 y enfermedad cardiovascular. Revista Colombiana de Cardiología, 27(3). <https://www.elsevier.es/es-revista-revista-colombiana-cardiologia-203-articulo-covid-19-enfermedad-cardiovascular-S0120563320300760>

Urbina, I. & El Hospital. (2016, junio). Panorama del mercado de la industria cardiovascular. El Hospital. Recuperado 13 de noviembre de 2021, de <https://www.elhospital.com/temas/Panorama-del-mercado-de-la-industria-de-tecnologias-para-cardiologia+113162>

KiCad EDA. (2021, 28 mayo). About KiCad. KiCad Website. Recuperado 22 de julio de 2022, de <https://www.kicad.org/about/kicad/>

MDN contributors. (2020, 8 diciembre). Mensajes HTTP - HTTP | MDN. MDN Web Docs.  
<https://developer.mozilla.org/es/docs/Web/HTTP/Messages>

Red Hat. (s. f.). ¿Qué es una API de REST? Red Hat | Líder mundial en código abierto.  
Recuperado 22 de julio de 2022, de  
<https://www.redhat.com/es/topics/api/what-is-a-rest-api>

Bravo, L. (2018, 5 septiembre). Framework o librerías: ventajas y desventajas. tiThink  
Technology. Recuperado 4 de agosto de 2022, de  
<https://www.tithink.com/es/2018/08/29/framework-o-librerias-ventajas-y-desventajas/>

Mayo Clinic. (s. f.). Taquicardia. Recuperado de  
<https://www.mayoclinic.org/es-es/diseases-conditions/heart-arrhythmia/symptoms-causes/syc-20350668#:~:text=La%20taquicardia%20es%20un%20latido,a%2060%20latidos%20por%20minuto.>

Apéndices

Apéndice A. Documento plan de pruebas ritmo cardíaco y saturación

**Plan de pruebas proyecto: Diseño de una plataforma IoT para el monitoreo y registro de variables en pacientes con problemas cardiovasculares bajo cuidados domiciliarios.**

Consideraciones antes de iniciar las pruebas

- Se debe ubicar y fijar correctamente el dispositivo al paciente, para cada prueba
- El paciente no debe realizar movimientos bruscos que puedan desprender el dispositivo
- Se debe configurar previamente la conexión entre el dispositivo y el teléfono celular
- Se deben preparar los dispositivos médicos necesarios para llevar a cabo las pruebas

**Pruebas**

Fecha de ejecución de pruebas: 17-05-23

- Se toma el ritmo cardíaco durante 1 minuto con el dispositivo médico Pulsioxímetro obteniendo y con el prototipo obteniendo

Toma	Dispositivo[bpm]	Prototipo [bpm]	Observación
1	74	77	
2	66	67	
3	66	69	
4	59	60	
5	64	63	
	71	70	

- Se toma el porcentaje de oxígeno durante 1 minuto con el dispositivo médico pulsioxímetro obteniendo y con el prototipo obteniendo

Toma	Dispositivo[spO2]	Prototipo [spO2]	Observación
1	95	99	
2	94	99	
3	95	99	
4	98	99	
5	97	99	

Apéndice B. Documento plan de pruebas temperatura y observaciones

- Se toma la temperatura con el dispositivo médico Termómetro, obteniendo y con el prototipo obteniendo

Toma	Dispositivo[°C]	Prototipo [°C]	Observación
1	36	36	
2	36	35	
3	35	35	
4	35	36	
5	35	37	

- Se toman 2 registros de pulsaciones de 30 segundos con el dispositivo médico pulsioxímetro y con el prototipo, se almacenan imágenes para comparación (Adjuntos)

- Se prueban registros en la página web y en la aplicación, con la supervisión de la Dra. Laura Isabel Valencia Angel recibiendo los comentarios de la misma, un punto importante es verificar la frecuencia con la que se toman los datos

Uno de los puntos importantes respecto a la toma de datos es el tiempo que tarda el dispositivo en adquirir una cantidad de datos considerable como para presentar mediciones precisas, el cual se observó que era alrededor de 30 segundos desde el inicio de cada prueba.  
Se lograron tomar 12 registros exitosos con el dispositivo enlazado a la aplicación móvil y reportados también a la plataforma web.

**Observaciones**

Dispositivo (Comodidad, usabilidad, tamaño, diseño, sensor):

Se sugirió por parte de los voluntarios que hicieron parte de la prueba, que el dispositivo tuviese un tamaño menor y redistribuir la ubicación de los sensores, de tal manera que no dificultara realizar ciertas actividades diarias.  
También se recomendó el uso de una batería portátil para no verse limitado por una fuente de poder externa.

## Apéndice C. Documento plan de pruebas observaciones

Página web (Usabilidad, diseño, representación de datos, información mostrada, precisión datos):

Los participantes sugirieron la creación de diferentes tipos de alerta, de tal forma que permitiera así priorizar a varios pacientes que los hayan generado.  
También hablaron sobre la posibilidad de tener acceso a más información de cada uno de los pacientes, como el sistema de información del hospital.

Aplicación (Usabilidad, diseño, representación de datos, información mostrada, alertas)

Puede ser innecesario mostrar la representación gráfica de las mediciones cardíacas, puesto que la aplicación estaría siendo usada por el paciente o encargado y no serían de utilidad para éstos.  
Es positivo que se muestre una alerta en la aplicación de tal forma que si se ve interrumpida la conexión a internet, se pueda alertar al personal médico por otros medios.

Adicionales:

Se sugirió como trabajo futuro, probar otros sensores que se pudiesen ajustar más a las necesidades del proyecto, por ejemplo usar un sensor de temperatura de contacto, en vez de un infrarrojo, así como también evaluar el uso de un sensor de pulsioximetría de mayor precisión de cara a un uso médico.