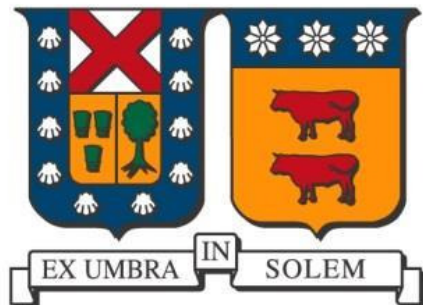


Universidad Técnica Federico Santa María

**DEPARTAMENTO DE INFORMÁTICA
VALPARAISO - CHILE**



“Sistema integrado de monitoreo de pacientes”

DAMIAN NICOLAS ASCUI AGUILERA

**MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA**

**Profesor Guía: José Lino Contreras Véliz
Profesor Correferente: Eda Arlene Siches Bahamondes**

Mes Junio - 2023

DEDICATORIA

Me complace dedicar la presente tesis a todas las personas que han sido fundamentales en mi camino y que han brindado su apoyo incondicional a lo largo de estos años.

En primer lugar, quiero expresar mi profundo agradecimiento a mi familia. Su amor, aliento y apoyo han sido fundamental a lo largo de todo mi viaje académico. A mi profesor guía, agradezco sinceramente su guía y dedicación a lo largo de todo el proceso de investigación. Sus conocimientos y consejos han sido vitales para el desarrollo de este trabajo. Agradezco también a mis amigos y compañeros de estudio, quienes me han acompañado en cada etapa de este recorrido académico. Y finalmente, quiero dedicar un agradecimiento a mi pareja. Por ser un gran apoyo, durante toda esta travesía, por tu comprensión y amor incondicional. Gracias por estar a mi lado en momentos difíciles.

Espero que este trabajo contribuya de alguna manera al avance del conocimiento.

Con cariño,

Damian Ascui

RESUMEN

Resumen—En el presente documento se expondrá como se ha realizado el diseño y desarrollo de un sistema integrado de monitoreo de pacientes. El trabajo realizado se segmenta en tres partes principales: la obtención de datos sobre soluciones actuales y diferentes dispositivos que obtienen los signos vitales del paciente, propuesta y evaluación de factibilidad de estas, finalmente desarrollo del sistema. El sistema desarrollado cumple los objetivos y propósitos presentados, y da la posibilidad de ampliar la arquitectura presentada y sus funcionalidades o la difusión de esta solución en ambientes rurales o de difícil acceso para funcionarios de la salud.

Palabras Clave— Biosensores; IOT; Integración de tecnologías; signos vitales.

ABSTRACT

Abstract—This document will present the design and development of an integrated patient monitoring system that has been accomplished. The work done is segmented into three main parts, obtaining data on current solutions and different devices that obtain the patient's vital signs, proposal and evaluation of their feasibility, and finally development of the system. The developed system meets the objectives and purposes presented and gives the possibility of expanding the presented architecture and its functionalities or the dissemination of this solution in rural environments or places with difficult access for health officials.

Keywords—Biosensors; IOT; Integration of technologies; vital signs.

GLOSARIO

BT: Bluetooth

BD: Base de Datos

CRUD: Crear, Leer, Actualizar y Borrar (*create, read, update, and delete*)

DAO: Objeto de acceso a datos (*Data Access Object*)

ECG: Electrocardiograma

EPOC: Enfermedad pulmonar obstructiva crónica

GND: Tierra (*Ground*)

IoT: Internet de las Cosas (*Internet of Things*)

IR: Infrarroja

LO: Comparador *Leads Off*

SCL: Señal del reloj (*Serial Clock*)

SDA: Datos en serie (*Serial Data*)

VP: Sensor de voltaje positivo

INDICE DE CONTENIDOS

RESUMEN.....	3
ABSTRACT	3
ÍNDICE DE FIGURAS	8
INDICE DE TABLAS	12
INTRODUCCIÓN.....	13
CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA	14
1.1 El Mercado actual de sistema de monitoreo de pacientes.....	15
1.2 Vista al Futuro.....	15
1.3 El Mercado y Actores en Chile	17
1.4 Otros dispositivos de monitoreo de pacientes.....	18
1.4.1 Monitores de signos vitales y sensores biológicos actuales.....	18
1.4.2 Relojes inteligentes.....	18
1.4.3 E-Health Sensor Shield V2.0	19
CAPÍTULO 2: MARCO CONCEPTUAL	20
2.1 Telemedicina y Telesalud	20
2.1.1 Telesalud.....	20
2.1.2 Telemedicina.....	20
2.2 El Internet de las Cosas (IoT).....	22
2.3 Signos Vitales y sus mediciones	22
2.3.1 Respiración/Frecuencia Respiratoria.....	22
2.3.2 Espirómetro.....	24
2.3.3 Medición Manual.....	24
2.3.4 Temperatura Corporal.....	24
2.3.5 Termómetros.....	25
2.3.6 Frecuencia Cardíaca.....	25
2.3.7 Electrocardiógrafo	27

2.3.8 Tensión Arterial	28
2.3.9 Tensiómetro	29
2.4 Tecnologías a implementar	30
2.4.1 Sensores	30
2.4.2 ESP32	31
2.4.3 Max30102:.....	31
2.4.4 Bluetooth	33
2.4.5 AD8232 Sensor de ECG	33
2.4.6 Android	34
2.4.7 Android Studio.....	34
2.4.8 OpenCV	35
2.4.9 Firebase	36
CAPÍTULO 3: PROPUESTA DE SOLUCIÓN	37
3.1 Componentes para crear el sistema:.....	39
CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN	41
4.1 Análisis de Requerimientos	41
4.1.1 Requerimientos Funcionales	41
4.1.2 Requerimientos no Funcionales	42
4.2 Descripción de Casos de uso	42
4.3 Diagrama de Casos de uso	46
4.4 Diagramas de Secuencia.....	50
4.5 Diagrama de Clases.....	58
4.6 Diagrama de Actividades.....	59
4.7 Diagrama de Estados	60
4.8 Diseño y descripción de interfaces.....	61
4.8.1 Interfaz de <i>Login</i> y <i>Sign Up</i>	61
4.8.2 Interfaz <i>Dashboard</i>	62
4.8.3 Interfaz Añadir Mediciones	64

4.9 Base de Datos Firebase	65
4.10 Implementación de <i>Login/Sign Up</i>	67
4.11 Implementación de funciones	68
4.11.1 Afiliar Usuarios	68
4.11.2 Ver Registros	69
4.12 Implementación Base de Datos.....	70
4.13 Implementación Reconocimiento de Imágenes.....	71
4.14 Implementación de <i>Dashboard</i>	74
4.15 Implementación de comunicación Bluetooth	75
4.16 Implementación Max30102.....	76
4.17 Implementación ADB232	77
CAPÍTULO 5: CONCLUSIONES.....	78
REFERENCIAS BIBLIOGRÁFICAS.....	82
ANEXOS.....	86

ÍNDICE DE FIGURAS

Figura 1 :Monitor Signos Vitales, Sera Medica, https://ventadeequipomedico.mx/tienda/equipo-de-quiroyano/monitores-para-signos-vitales/monitor-signos-vitales-basico/	18
Figura 2: Reloj Inteligente Smartwatch M4 Deportivo Pulsera Deportes, https://www.falabella.com/falabella-cl/product/117138443/Reloj-Inteligente-Smartwatch-M4-Deportivo-Pulsera-Deportes/117138444	19
Figura 3: e-Health Sensor Platform for Arduino and Raspberry Pi, https://www.openhacks.com/page/productos/id/1605/title/e-Health-Sensor-Platform-for-Arduino-and-Raspberry-Pi	19
Figura 4: "Aparato Respiratorio", Lifeder, https://www.lifeder.com/aparato-respiratorio/	23
Figura 5: Espirómetro digital Vitalograph COPD-6, GirodMedical, https://www.girodmedical.es/blog_es/espironetro-para-que-sirve/	24
Figura 6: Termómetro de tubo de vidrio y digital, elEconomista, https://www.economistaamerica.cl/actualidad-eAm-chile/noticias/10444510/03/20/Que-hacer-si-se-rompe-el-termometro-en-casa.html	25
Figura 7: Partes del Corazón, Fenómeno Eléctrico, Anatomía y fisiología del cuerpo humano.	26
Figura 8: Ciclo cardiaco normal (A) y sistema de conducción del corazón (B), Pediatr Integral 2012; XVI(9): 715-722, https://www.pediatriaintegral.es/numeros-anteriores/publicacion-2012-11/lectura-del-ecg/	27
Figura 9:Electrocardiografo, MedicalTec, https://medicaltec.cl/shop/electrocardiografos	27
Figura 10: Aneroide (A) y tensiómetro digital (B), https://hiraoka.com.pe/blog/post/tensiometro-que-es-para-que-sirve-y-como-funciona	29
Figura 11: Sensor de pulso y Oxímetro MAX30102, https://www.mechatronicstore.cl/sensor-de-pulso-y-oximetro-max30102-arduino/	31
Figura 12: Detección de pulso max30102, https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/	32
Figura 13: Detección del pulso sensor de frecuencia cardíaca, https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/	32
Figura 14: Espectro de absorción de la hb y la hbo2, https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/	33
Figura 15: AD8232 ECG Sensor, How to Electronics, https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/	33
Figura 16: Esquema de Arquitectura, Xataka Android, https://www.xatakandroid.com/sistema-operativo/que-es-android	34

Figura 17: Diagrama de Hardware de solución Inicial, Elaboración Propia.....	38
Figura 18: Diagrama de Hardware de solución, Elaboración Propia	38
Figura 19: Diagrama caso de uso Login, Elaboración Propia.....	46
Figura 20: Diagrama caso de uso Conexión BT, Elaboración Propia.....	47
Figura 21: Diagrama caso de uso Registrar usuarios, Elaboración Propia	47
Figura 22: Diagrama caso de uso Modificar mediciones, Elaboración Propia	48
Figura 23: Diagrama caso de uso Registrar mediciones, Elaboración Propia	48
Figura 24: Diagrama caso de uso Modificar usuarios, Elaboración Propia	49
Figura 25: Diagrama caso de uso Ver datos, Elaboración Propia	49
Figura 26: Diagrama caso de uso Afiliar usuarios, Elaboración Propia.....	50
Figura 27: Diagrama de secuencia Registro, Elaboración Propia	51
Figura 28: Diagrama de secuencia Registro mediciones, Elaboración Propia.....	51
Figura 29: Diagrama de secuencia Login, Elaboración Propia.....	52
Figura 30: Diagrama de secuencia Conexión BT, Elaboración Propia.....	53
Figura 31: Diagrama de secuencia Modificar Usuario, Elaboración Propia	54
Figura 32: Diagrama de secuencia Ver datos, Elaboración Propia	55
Figura 33: Diagrama de secuencia Ver datos, Elaboración Propia	55
Figura 34: Diagrama de secuencia Modificar mediciones, Elaboración Propia.....	56
Figura 35: Diagrama de secuencia Afiliar usuario, Elaboración Propia.....	57
Figura 36: Diagrama de clases, Elaboración Propia.....	58
Figura 37: Diagrama de actividades, Elaboración Propia.....	59
Figura 38: Diagrama de estados, Elaboración Propia	60
Figura 39: Interfaz de Bienvenida, Elaboración Propia.....	61
Figura 40: Interfaz de Registro, Elaboración Propia	61
Figura 41: Interfaz de Login, Elaboración Propia.....	61
Figura 42: Interfaz de Conexión BT, Elaboración Propia.....	63
Figura 43: Interfaz de Configuración, Elaboración Propia	63
Figura 44: Interfaz de Ver registro, Elaboración Propia.....	63

Figura 45: Interfaz de Afiliados, Elaboración Propia	63
Figura 46: Interfaz de Dashboard Medico, Elaboración Propia	63
Figura 47: Interfaz de <i>Dashboard</i> , Elaboración Propia	63
Figura 48: Interfaz de añadir foto, Elaboración Propia.....	64
Figura 49: Interfaz de <i>cropeo</i> de foto, Elaboración Propia	64
Figura 50: Interfaz de añadir manual, Elaboración Propia	64
Figura 51: Conexión Android – Firebase , https://console.firebase.google.com	65
Figura 52: Base de datos Firebase, https://console.firebase.google.com	66
Figura 53: Código sincronización inicial, Elaboración Propia	66
Figura 54: Código Login, Elaboración Propia	67
Figura 55: Código SignUp, Elaboración Propia	67
Figura 56: Código buscar usuarios para afiliar, Elaboración Propia	68
Figura 57: Código RecyclerView, Elaboración Propia	69
Figura 58: Código SwipeHelper, Elaboración Propia	69
Figura 59: Código Dao Medición, Elaboración Propia	70
Figura 60: Código Conversión, Elaboración Propia	70
Figura 61: Código añadir medición, Elaboración Propia	70
Figura 62: Procedimiento OpenCV marcado de números, Elaboración Propia	71
Figura 63: Número de siete segmentos, https://en.wikipedia.org/wiki/File:7_segment_display_labeled.svg	72
Figura 64: Lista códigos de cada número, Elaboración Propia	72
Figura 65: Código Orden de números, Elaboración Propia	73
Figura 66: Código algunas transformaciones de la imagen, Elaboración Propia	73
Figura 67: Código <i>GetUsers UserbyId</i> , Elaboración Propia	74
Figura 68: Código <i>Thread</i> receptora de datos, Elaboración Propia.....	75
Figura 69: Max30102 Pins Layout, https://1stminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/	76
Figura 70: Parte del Código ESP32 Max30102, Elaboración Propia.....	77

Figura 71: Conexiones con la ESP32, <https://www.circuitschools.com/ecg-monitoring-system-using-ad8232-with-arduino-or-esp32-iot-based/> 77

Figura 73: Código ADB232 Parte 2, Elaboración Propia 78

Figura 72: Código ADB232 Parte 1, Elaboración Propia 78

INDICE DE TABLAS

Tabla 1: Descripción caso de uso <i>login</i> , Elaboración Propia.....	42
Tabla 2: Descripción caso de uso <i>bluetooth</i> , Elaboración Propia.....	42
Tabla 3: Descripción caso de uso registró mediciones, Elaboración Propia	43
Tabla 4: Descripción caso de uso registrar datos, Elaboración Propia	43
Tabla 5: Descripción caso de uso datos biosensor, Elaboración Propia.....	43
Tabla 6: Descripción caso de uso modificar usuario, Elaboración Propia	44
Tabla 7: Descripción caso de uso modificar mediciones, Elaboración Propia	44
Tabla 8: Descripción caso de uso <i>dashboard</i> , Elaboración Propia	44
Tabla 9: Descripción caso de uso afiliar, Elaboración Propia.....	45
Tabla 10: Descripción caso de uso ver datos, Elaboración Propia	45
Tabla 11: Descripción caso de uso ver datos, Elaboración Propia	45
Tabla 12: Descripción caso de uso base de datos nube, Elaboración Propia	46
Tabla 13: Descripción caso de uso registrar usuario, Elaboración Propia.....	46

INTRODUCCIÓN

En los últimos años, el sector sanitario ha evolucionado rápidamente con la integración de nuevas tecnologías para mejorar la atención y el tratamiento clínico de los pacientes. Uno de estos avances tecnológicos es el sistema integrado de monitorización de pacientes. Este sistema implica el uso de diversos sensores y dispositivos que controlan y registran las constantes vitales del paciente y otros datos relevantes, proporcionando información en tiempo real a los profesionales de los servicios de salud. Esta información puede utilizarse para detectar signos tempranos de deterioro del estado del paciente, lo que permitiría intervenciones oportunas y prevenir resultados adversos.

En este trabajo se busca desarrollar un prototipo de un sistema integrado de monitoreo de pacientes en base a componentes informáticos de bajo costo. La tesis examinará el estado actual del seguimiento de pacientes, las tecnologías implicadas y el impacto potencial en los resultados de los pacientes. Además se analizarán los retos asociados con la implementación de sistemas de monitorización integrados, incluyendo la gestión de datos y los problemas de privacidad, las limitaciones técnicas y la necesidad de formación y educación del personal.

La tesis contiene una combinación de revisiones bibliográficas, estudios y entrevistas a expertos para proporcionar un análisis sobre el impacto potencial en la atención al paciente del sistema integrado de monitorización de pacientes.

Este texto muestra los distintos aspectos del desarrollo de un sistema integrado de monitoreo de pacientes. En el primer capítulo se presenta la problemática a la cual responden estos sistemas, además del mercado actual en Chile y el mundo, junto con otros dispositivos que pueden cumplir las funciones de monitoreo de pacientes. En el segundo capítulo se detallan conceptos, términos y tecnologías que se utilizarán para el desarrollo del sistema. En el tercer capítulo se presentan los objetivos que se buscan obtener, además de la estructura del sistema final, respondiendo a las problemáticas mencionadas en el primer capítulo. En el cuarto capítulo se presenta la solución, a través de, un análisis de requerimientos, diagramas para generar la solución y descripciones de cada una de las partes del sistema. El último capítulo corresponde a las conclusiones del sistema desarrollado e informe de memoria.

En general, el sistema integrado de monitorización de pacientes representa una oportunidad significativa para mejorar la seguridad y los resultados de los pacientes, y esta tesis pretende aportar ideas sobre los desafíos, problemáticas y soluciones que puede encontrar durante el diseño y desarrollo de este tipo de sistemas.

CAPÍTULO 1: DEFINICIÓN DEL PROBLEMA

En algunas ciudades, principalmente con alta densidad de población, los centros hospitalarios cuentan con salas de atención equipadas con equipos médicos especializados para satisfacer la demanda de atención médica. Sin embargo, surge un problema cuando las personas no pueden acudir a estos centros médicos por diversas razones, como la falta de transporte público o privado, pacientes postrados o inmunodeprimidos. En estos casos, se requiere adquirir equipos de monitoreo de signos vitales que permitan supervisar constantemente la salud del paciente, sin necesidad de contratar personal médico capacitado. Es fundamental que este sistema sea fácil de usar, de manera que los enfermeros o asistentes puedan controlar al paciente, visualizar su estado actual y tomar decisiones de manera oportuna para su cuidado.

El cuerpo, mediante diversos signos vitales que manifiesta a lo largo del día, es una fuente de información permanente que indica su estado de salud, entendiéndose como las condiciones médicas (salud física y mental), experiencias en reclamaciones, admisión de cuidado de salud, historia clínica, información genética, evidencia de asegurabilidad e incapacidad (CuidadoDeSalud.gov, n.d.). Sin embargo, pocas veces accedemos a esta información de manera oportuna, y en general se busca una vez que se ha instaurado un problema en nuestro organismo, situación que para este momento podría ser muy tarde.

La prevención es la principal respuesta a esta problemática pues logra minimizar el impacto o prevenir la aparición de alguna enfermedad, como la presencia de factores de riesgo, y así detener su avance y atenuar sus consecuencias si la enfermedad ya está presente. A pesar de las ventajas de la prevención, *“solo un 13% de las mujeres de entre 20 y 65 años, y un 16% en los hombres del mismo rango etario, inscritos en los consultorios de nuestro país, realizaron el examen de Medicina Preventiva (EMP) durante el año 2018.”* (Leiva, 2019)

La falta de información oportuna es más determinante en el caso de enfermedades crónicas que requieren especial cuidado por parte del paciente; pues según datos ofrecidos por la Organización Mundial de la Salud (OMS), *“las enfermedades crónicas no transmisibles son responsables del 60% de las muertes en nuestro planeta. Pero se estima que su progreso podría ser controlado, o por lo menos reducido, si se aplicarán programas específicos de prevención de la salud a escala global”*. (OKDIARIO, 2019)

Por lo tanto, resulta de vital importancia disponer de un sistema que sea capaz de resolver esta problemática, entregando información confiable y actualizada, de forma que la atención de salud pueda darse oportunamente, reduciendo de esta forma las complicaciones que se puedan tener en estado de emergencia.

1.1 El Mercado actual de sistema de monitoreo de pacientes

Según los informes de investigación global sobre el 'Mercado de dispositivos digitales de monitoreo de pacientes', el mercado del monitoreo ha crecido exponencialmente en los últimos dos años y se espera que esta tendencia continúe siguiendo equivalente hasta 2026 (Lawson, 2021).

Dispositivos como medidores portables y tecnologías en prendas de vestir, como *Smart Watches*, están siendo cada vez más aceptadas por la industria de la salud, a medida que estos son cada vez más capaces de llevar registro de diferentes parámetros de manera continua y ofrecer información valiosa a médicos, en cualquier momento, sin la necesidad de visitas clínicas (Surabhi Pandey, 2020).

Aun así, los pacientes se quejan de ciertos inconvenientes relacionados con el uso de este tipo de dispositivos, además de su alto costo, restringiendo la disponibilidad a solo algunos pacientes. Por todo ello, los investigadores y empresas de *MedTech* (Empresas en el rubro de la creación de sensores biométricos) continúan buscando soluciones innovadoras para la monitorización de pacientes sin contacto, en respuesta a las necesidades insatisfechas del mercado actual (Finnegan, 2020).

En la lucha contra el COVID-19 las aplicaciones de salud y sistemas de monitoreo sin contacto han sido parte vital, y de esto nacen estudios como *Frost & Sullivan, Advanced Non-Contact Patient Monitoring Technologies: A New Paradigm in Healthcare Monitoring* (Surabhi Pandey, 2020).

En este estudio se destaca que, si bien estas aplicaciones incorporan tecnologías como algoritmos de inteligencia artificial para el aprendizaje automático, las últimas tecnologías de monitoreo sin contacto gravitan hacia el uso de video, análisis de sonido y plataformas móviles.

1.2 Vista al Futuro

"La industria de la salud está cambiando su enfoque hacia soluciones de monitoreo remoto basadas en valores y centradas en el paciente, proporcionando un estímulo para la investigación de tecnologías de monitoreo sin contacto", explicó Ashish Kaul, analista de investigación *Senior* de *Technical Insights*. "Las tecnologías sin contacto son discretas, rentables y pueden usarse para monitorear a múltiples usuarios, lo que las convierte en una solución efectiva para monitorear a los pacientes en masa" (Frost & Sullivan, 2020)

El experto agregó que "la actual situación pandémica del COVID-19 ofrece una gran oportunidad para las tecnologías de monitoreo de pacientes sin contacto, ya que facilitan la detección temprana de complicaciones respiratorias en pacientes sospechosos; su adopción reduce las posibilidades de infección entre médicos y trabajadores de la salud; y, además, estas tecnologías pueden ser utilizadas por las empresas farmacéuticas para

monitorear los resultados de los fármacos desarrollados por ellas ” (Frost & Sullivan, 2020).

Dentro de las tecnologías que se espera que tengan un impacto para mercado futuro se encuentran:

- Tecnologías de análisis de sonido para la monitorización remota de pacientes con enfermedades similares a la gripe (ILI) en función de su tos.
- Tecnologías de monitoreo sin contacto basadas en video que pueden diagnosticar el estrés mental al detectar signos fisiológicos y emocionales como depresión, enojo e inquietud.
- Tecnologías de sensores de radar capaces de rastrear la respiración y la frecuencia cardíaca para detectar síntomas de enfermedades respiratorias como asma, EPOC y COVID-19.
- Tecnologías sin contacto basadas en sensores capaces de rastrear la frecuencia cardíaca, la frecuencia respiratoria y la presión arterial para las personas interesadas en su salud y estado físico.

Finalmente, cabe destacar que la inteligencia artificial (IA) ha transformado rápidamente muchas industrias en los últimos años, y el campo de la medicina no es una excepción. Varios expertos creen que el próximo año marcará el comienzo de un mayor impulso de la atención médica personalizada, ayudado por el aumento de los volúmenes de datos en el sector. IDC predice que *"la esfera de información mundial pasará de 33 zettabits de datos en 2018 a 175 zettabits en 2025"*, lo que supone multiplicar por cinco las cifras de 2018. (Adebayo, 2022)

Conforme pasa el tiempo, los profesionales de la IA siguen analizando las tendencias de los años presentes con el fin de predecir que nos deparará en el futuro. Venture Beat conversó con varios de estos expertos para obtener una visión de hacia dónde podría dirigirse el campo de la inteligencia artificial en el ámbito médico:

- La atención sanitaria personalizada será aún más importante.
- La legislación y la normativa en torno a la IA sanitaria mejorarán.
- Se redoblarán los esfuerzos para combatir el sesgo de la IA.
- Una gama más amplia de aplicaciones en sanidad.
- Una relación de trabajo más estrecha entre los humanos y la IA.
- Automatización.

Aunque se reconoce que llevará tiempo antes de que los sistemas de IA puedan reflejar la empatía que influye en muchas decisiones humanas, esto no implica que no deban seguir mejorándose constantemente para imitar de manera más precisa los valores humanos. *"La IA sólo refleja la programación y los datos que la componen, y los directivos de las empresas deben ser conscientes de que las ideas frías y basadas en datos son sólo una parte del proceso total de toma de decisiones"*, afirma Andy Thurai, vicepresidente y analista principal de Constellation Research. Sin duda, el año que viene aumentará el uso de la IA en la medicina, a medida que surjan más aplicaciones y las organizaciones empiecen a dar prioridad a la IA responsable. En palabras de Laster, *"en general, el futuro de la IA en la sanidad parece prometedor"*. (Adebayo, 2022)

1.3 El Mercado y Actores en Chile

En nuestro país, el avance general en telemedicina de los últimos años sigue la misma línea de desarrollo que países desarrollados, como Estados Unidos, debido a la pandemia que ha vivido nuestro país y el resto del mundo, la telemedicina ha avanzado junto con nuevas herramientas tecnológicas, el impacto y desarrollo ya ha sido estudiado por organismos como el Centro Nacional en Sistemas de Información en Salud (CENS), el Hospital Clínico de la Universidad de Chile, la Red de Salud UC Christus, el Colegio Médico, ACHISA, entre otros. En una entrevista que se realizó con Viviana Torres, líder de Informática Clínica del CENS, en ésta afirma que *"La telemedicina en el último año ha visto un desarrollo exponencial a nivel mundial"*, sostiene Torres agregando que Chile no es la excepción. (Torres V. , 2020)

Un ejemplo que Torres destaca es el desarrollo del Hospital Digital, iniciativa del Ministerio de Salud de Chile¹, otros avances que se pueden destacar son la Guía de Buenas Prácticas para Telemedicina², cuya versión final fue desarrollada junto a la OMS, la OPS, el CDC de Estados Unidos, y el NHS de Reino Unido, siendo publicada el 14 de abril de 2020 ; también se destaca la Estación Monitoreo Remoto de Pacientes COVID-19, publicado el 12 de mayo de 2020 por la Universidad de Chile, que busca capturar los signos vitales de los pacientes, que son relevantes para la enfermedad, desde el hogar (Universidad de Chile , 2020).

Viviana Torres comenta, *"Hoy existe un porcentaje no menor de emprendimientos y empresas que están innovando en torno a la salud digital, creando herramientas muy útiles que permiten potenciar la telemedicina. Por ejemplo, algunas de las herramientas son Snabb, quienes trabajan con el agendamiento de horas médicas, o Alumbra Toraxia, quienes promueven el análisis de radiografías mediante la inteligencia artificial"*, muestra de un auge en las tecnologías enfocadas a fortalecer las actuales herramientas de la telemedicina del país. (Torres V. , 2020)

¹ <https://saluddigital.com/es/el-mundo-en-la-nube/el-hospital-digital-una-iniciativa-del-ministerio-de-salud-de-chile-para-hacer-frente-a-la-nueva-carga-de-enfermedad/>

² <https://cens.cl/wp-content/uploads/2020/04/cens-gua-teleconsulta-en-tiempos-de-pandemia.pdf>

1.4 Otros dispositivos de monitoreo de pacientes

1.4.1 Monitores de signos vitales y sensores biológicos actuales



Figura 1 :Monitor Signos Vitales, Sera Medica, <https://ventadeequipomedico.mx/tienda/equipo-de-quirofano/monitores-para-signos-vitales/monitor-signos-vitales-basico/>

Los Monitores de signos vitales corresponden a dispositivos capaces de medir los principales signos que emite el cuerpo que demuestran su funcionamiento, como son el pulso cardíaco, temperatura o presión arterial. Comúnmente estas medidas son parte de un chequeo médico, donde se obtiene información del paciente mediante estos.

Por otro lado, los sensores biológicos, corresponden a instrumentos que permiten medir parámetros químicos o biológicos, se constituyen normalmente de tres partes, una muestra que puede ser un tejido, un cultivo de microorganismos, enzimas, anticuerpos, cadenas de ácidos nucleicos, etc., un detector que al momento de ponerse en contacto con el elemento biológico ocurrirá un cambio físico o químico en el material, finalmente un traductor que

emitirá una señal eléctrica en función del cambio ocurrido en el detector pueden llegar a ser más invasivos y realizan una medición más selectiva.

Estos representan una solución parcial al problema ya que permiten conseguir la información que se desee, ya sea el pulso, oxígeno en la sangre, temperatura, cantidad de glucosa en la sangre, presión arterial, etc. Sin embargo, esta solución actualmente no proporciona una comunicación directa con el médico con el que se está tratando ni una central donde se puedan revisar y almacenar todos los datos que se obtiene de estos, por lo que sigue siendo una complicación para el usuario tener que llevar un registro de todas las mediciones que se haga, además enfermeros y médicos no tienen forma de detectar cambios de manera inmediata en sus pacientes o revisar los que se han producido a lo largo del tiempo en pacientes que no pertenezcan a sus registros, pudiendo ser indicativo de alguna enfermedad o deterioro del estado de salud del paciente.

1.4.2 Relojes inteligentes

Estos dispositivos actualmente cuentan con sensores básicos, como por ejemplo la frecuencia cardíaca, y permite obtener estadísticas, además de alarmas en caso de que alguna de las mediciones llegue a un nivel muy elevado o demasiado bajo, sin embargo, no son capaces de utilizar sensores más complejos y avanzados como por ejemplo los Electrocardiogramas o Glucómetro. (Weiss, 2019)



Figura 2: Reloj Inteligente Smartwatch M4 Deportivo Pulsera Deportes, <https://www.falabella.com/falabella-cl/product/117138443/Reloj-Inteligente-Smartwatch-M4-Deportivo-Pulsera-Deportes/117138444>

1.4.3 E-Health Sensor Shield V2.0

E-Health Sensor Shield V2.0 es una placa permite a los usuarios de microcomputadores Arduino y Raspberry Pi realizar aplicaciones biométricas y médicas en las que sea necesaria la monitorización del cuerpo mediante el uso de 10 sensores diferentes: pulso, oxígeno en sangre (SPO2), flujo de aire (respiración), temperatura corporal, electrocardiograma (ECG), glucómetro, respuesta galvánica de la piel (GSR - sudoración), presión sanguínea (esfigmomanómetro), posición del paciente (acelerómetro) y sensor muscular/electromiográfico (EMG).



Figura 3: e-Health Sensor Platform for Arduino and Raspberry Pi, <https://www.openhacks.com/page/productos/id/1605/title/e-Health-Sensor-Platform-for-Arduino-and-Raspberry-Pi>

Esta información puede utilizarse para controlar en tiempo real el estado de un paciente u obtener datos sensibles para su posterior análisis con fines de diagnóstico médico. La información biométrica recopilada puede enviarse de forma inalámbrica utilizando cualquiera de las 6 opciones de conectividad disponibles: Wi-Fi, 3G, GPRS, Bluetooth, 802.15.4 y ZigBee en función de la aplicación. (soloelectronicos, 2017)

CAPÍTULO 2: MARCO CONCEPTUAL

En este capítulo se detallan los fundamentos teóricos que apoyan la investigación, proporcionando una base para comprender el funcionamiento de las tecnologías y conceptos en que se desarrollara el sistema.

2.1 Telemedicina y Telesalud

Telemedicina y telesalud suelen utilizarse indistintamente, pero no son exactamente lo mismo. La telemedicina se refiere específicamente al uso de la tecnología para prestar servicios clínicos a distancia a los pacientes. Puede incluir consultas virtuales con profesionales de la salud, seguimiento a distancia de los datos sanitarios del paciente y transmisión de imágenes médicas y resultados de pruebas.

Por otra parte, telesalud es un término más amplio que abarca no sólo los servicios clínicos a distancia, sino también los servicios no clínicos que utilizan la tecnología para apoyar la prestación de asistencia sanitaria. Esto puede incluir cosas como la educación médica en línea, la monitorización remota de pacientes con enfermedades crónicas y el uso de aplicaciones sanitarias móviles para el seguimiento y la gestión de datos sanitarios.

2.1.1 Telesalud

Se define a la telesalud como al sistema que hace uso de las tecnologías de la información y comunicación para proporcionar servicios de salud, atención médica e información, independientemente de la distancia, teniendo foco en la prevención. Así también, la Telesalud es una herramienta para la difusión de información sobre el cuidado y prevención de enfermedades crónicas como las cardiovasculares, respiratorias, y por supuesto, las epidemias. (Barros, 2015)

2.1.2 Telemedicina

Por su parte, la telemedicina es una de las ramas más destacadas, conocidas y usadas de la Telesalud, pues de acuerdo con la Asociación Estadounidense de Telemedicina (*American Telemedicine Association, ATA*), *“La telemedicina es el intercambio de información médica entre dos actores que no están ubicados en el mismo espacio, que pueden ser médico/paciente, o médico/médico; a través de comunicaciones electrónicas, a través de algún medio de comunicación electrónico, con el objeto de mejorar el estado de salud de un paciente. Incluye una variedad cada vez mayor de aplicaciones y servicios que utilizan videoconferencias, correo electrónico, teléfonos inteligentes, comunicaciones inalámbricas y otras formas de tecnología de las telecomunicaciones”*. (Barros, 2015)

Se puede decir a partir de lo anterior que la telesalud es un sistema que hace uso de las tecnologías de la información y comunicaciones, pero tiene un carácter mayormente preventivo. (Barros, 2015)

Según la Organización Mundial de la Salud define telemedicina como: *“Suministro de servicios de atención sanitaria en los que la distancia constituye un factor crítico, realizado*

por profesionales que apelan a tecnologías de la información y de la comunicación con objeto de intercambiar datos para hacer diagnósticos, preconizar tratamientos y prevenir enfermedades y heridas, así como para: la formación permanente de los profesionales de atención de salud y en actividades de investigación y evaluación, con el fin de mejorar la salud de las personas y de las comunidades en que viven” (Brousse, 2006) Y “La distribución de servicios de salud, en el que la distancia es un factor crítico, donde los profesionales de la salud usan información y tecnología de comunicaciones para el intercambio de información válida para el diagnóstico, tratamiento y prevención de enfermedades o daños, investigación y evaluación; y para la educación continuada de los proveedores de salud pública, todo ello en interés del desarrollo de la salud del individuo y su comunidad” (Subdirección General de informática de INSALUD, Subdirección General de Sistemas y Tecnologías de la Información del Ministerio de Sanidad y Consumo, 2000).

Sin embargo, no se debe considerar la telemedicina únicamente como la trasmisión de información y comunicación sanitaria entre pacientes y médicos a distancia. Es, además, un medio de comunicación, formación y consulta entre profesionales de la salud, tanto del ámbito hospitalario como de atención primaria, y permite una mejor atención integral del paciente y formación continua de los profesionales de salud (Joseba Rabanales Sotosa, 2011).

La Telemedicina, por tanto, tiene como objetivo suministrar servicios de salud destinados a mantener el bienestar de la sociedad o mejorar su estado de salud general. Como servicio, comprende no sólo una prestación asistencial a los pacientes, sino también facilita los procesos administrativos y suministra información sanitaria. De esta forma, los usuarios de un sistema de Telemedicina pueden ser tanto los profesionales sanitarios (personal médico, de enfermería, administrativos, entre otros) como los pacientes y ciudadanos en general. (Brousse, 2006)

La literatura internacional identifica que los principales servicios de telemedicina son:

- Servicios de asistencia remota: pueden referirse, tanto a las teleconsultas de seguimiento, diagnóstico o tratamiento a distancia del paciente, como a los servicios de telemonitorización de pacientes –a menudo crónicos–, que incluyen en muchas ocasiones registros de parámetros biológicos.
- Servicios de gestión administrativa de las personas usuarias: incluyen, tanto la solicitud de exámenes, como los aspectos relacionados con la facturación por la prestación de servicios.
- La formación a distancia para profesionales, tiene como objetivo suministrar pautas y evidencias sobre salud que faciliten la educación continua de los profesionales de salud.
- La evaluación y la investigación colaborativa en red: el uso de las TICS para compartir y difundir buenas prácticas, así como crear conocimiento a través de las acciones y reacciones de sus miembros. (Subsecretaría de Redes Asistenciales, 2018)

2.2 El Internet de las Cosas (IoT)

El Internet de las cosas o IoT se define como la agrupación e interconexión de dispositivos y objetos a través de una red (privada o internet, red en red) donde todos los dispositivos y objetos pueden verse e interactuar. En cuanto al tipo de objetos o dispositivos, pueden ser cualquiera, desde sensores y dispositivos mecánicos hasta artículos de uso cotidiano como neveras, calzado o ropa. (Gracia, s.f.)

Proceso de implementar IoT permite conectar elementos físicos cotidianos al Internet: desde objetos domésticos comunes, como las ampolletas, hasta recursos para la atención de la salud, como los dispositivos médicos; también incluyen prendas y artículos personales, como los relojes inteligentes, e incluso los semáforos en ciudades inteligentes. Un sistema de IoT tradicional, como el hogar inteligente descrito anteriormente, funciona enviando, recibiendo y analizando datos de forma permanente en un ciclo de retroalimentación. Según el tipo de tecnología de IoT, las personas o la inteligencia artificial y el aprendizaje automático (IA/ML) utilizan esos datos para realizar análisis casi de inmediato o en cierto tiempo. (¿Qué es el Internet de las cosas (IoT)?, s.f.)

El internet de las cosas representa un futuro en el que los objetos cotidianos y el medio ambiente están conectados y controlados a través de una gama de dispositivos, redes de comunicación y servidores basados en la nube. Los componentes esenciales son los dispositivos de bajo costo, de bajo consumo y alto rendimiento. En el mercado, se espera que un gran número de dispositivos pueda ser integrado, ya que requieren puntos de precios sumamente bajos y bajo consumo de energía. Sin embargo, los dispositivos de gama alta, tales como vías de acceso, módulos de control de la máquina, visión inteligente, e incluso productos electrónicos de consumo son también importantes objetivos de crecimiento. (G. Wu, 2011)

2.3 Signos Vitales y sus mediciones

Ya que se debe realizar mediciones de los distintos signos vitales de una persona, es relevante comprender de donde provienen estos datos, es decir, se debe repasar algunos conceptos básicos del funcionamiento de los principales órganos y sistemas de nuestro organismo.

Primero se definirá la función general del órgano que nos entregará la medición, seguido de los datos relevantes que se esperan obtener, finalmente se explicará brevemente el funcionamiento de algunos aparatos que obtienen las mediciones buscadas.

2.3.1 Respiración/Frecuencia Respiratoria

El aparato respiratorio comprende la nariz, la faringe (garganta), la laringe (órgano de la voz), la tráquea, los bronquios y los pulmones (Figura 4). Sus partes se pueden clasificar de acuerdo con su estructura y función.

De acuerdo con su estructura, el aparato respiratorio consta de dos partes:

- El aparato respiratorio superior que abarca la nariz, la faringe y las estructuras asociadas
- El aparato respiratorio inferior que incluye la laringe, la tráquea, los bronquios y los pulmones.

Según su función, el aparato respiratorio también se puede dividir en dos partes:

- La zona de conducción que consiste en una serie de cavidades y tubos interconectados dentro y fuera de los pulmones, la nariz, la faringe, la laringe, la tráquea, los bronquios, que filtran, calientan y humectan el aire y lo conducen a los pulmones.
- La zona respiratoria que está constituida por tejidos dentro de los pulmones donde tiene lugar el intercambio gaseoso; los bronquios respiratorios, los conductores alveolares, los sacos alveolares y los alveolos, los sitios principales para el intercambio de gases entre el aire y la sangre (Tortora, 2020)

Aparato respiratorio

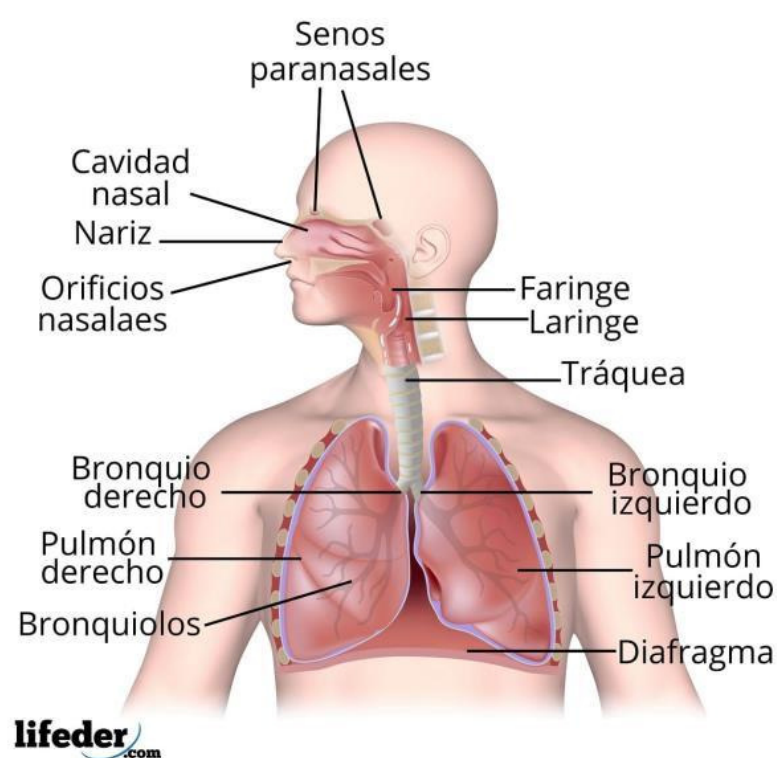


Figura 4: "Aparato Respiratorio", Lifeder, <https://www.lifeder.com/aparato-respiratorio/>

Medir la frecuencia respiratoria, consiste en determinar la cantidad de ciclos inspiración/espriación que se producen en el término de un minuto, observando el tórax de la persona para apreciar la profundidad de los movimientos. Además, conviene determinar las características de los movimientos respiratorios, consignando si son laboriosos, superficiales, profundos, etc. En condiciones normales, en un individuo adulto la respiración tiene una frecuencia que oscila entre 10 y 20 movimientos por minuto, es regular y silenciosa, y se desarrolla sin dificultad alguna. (Vay, 2019)



Figura 5: Espirómetro digital Vitalograph COPD-6, GirodMedical, https://www.girodmedical.es/blog_es/espirometro-para-que-sirve/

2.3.2 Espirómetro

El espirómetro es un aparato que calcula el volumen de aire espirado durante una respiración/inspiración forzada. Generalmente se utiliza para evaluar la capacidad pulmonar de sus pacientes. El objetivo es detectar cualquier anomalía respiratoria que pueda estar asociada a un problema de asma, problemas en los bronquios, o incluso debido al tabaco. (Girodmedical, 2017)

2.3.3 Medición Manual

La frecuencia respiratoria también se puede medir de manera manual, contando el número de veces que se respira en un minuto. Para tener una medición más precisa se debe:

- Mantener sentado y relajado.
- Se recomienda medir la frecuencia respiratoria sentado en una silla o en la cama.
- Mide la frecuencia respiratoria contando el número de veces en que el pecho, o el abdomen, se levanta durante un minuto.
- Registrar el número.

(C, 2020)

2.3.4 Temperatura Corporal

La temperatura interna del cuerpo humano se mantiene alrededor de los 37°C, lo cual es un requisito básico para su normal funcionamiento. Sin embargo, incluso en condiciones normales, los resultados de la medición variarán dependiendo de dónde se mida el cuerpo. La temperatura corporal puede ser anormal debido a fiebre (temperatura alta) o hipotermia (temperatura baja). Según la Asociación Médica Estadounidense, se considera que hay fiebre cuando la temperatura corporal es superior a 37 °C (98,6 °F) en la boca o 37,7 °C (99,8 °F) en el recto. La hipotermia se define como una temperatura corporal inferior a 95°F (35°C). (Redón, 2012)

2.3.5 Termómetros

Los termómetros son instrumentos utilizados para medir la temperatura, usando una escala graduada, mediante una sustancia que permita hacer las mediciones, al reaccionar con los cambios de temperatura, ya sea subiendo (calor) o bajando (frío). Para esto se toma un punto de partida conocido, que por lo común es el punto de fusión y ebullición de la sustancia, o por medio de los efectos eléctricos, por la dilatación de metales, o de otras índoles como los sensores térmicos, etc. Con los que se pueden observar los cambios de temperatura y medirlos con una escala. Algunos tipos de termómetros son los siguientes: (Tipos..., 2016)

- Termómetros de tubo de vidrio: Son termómetros que para realizar la medición de la temperatura tienen en su interior un líquido, (generalmente mercurio o alcohol), el cual se eleva conforme va aumentando la temperatura y decrece al ir bajando, y es mediante el nivel de este líquido que se toman las mediciones en una escala que previamente fue graduada.
- Termómetros digitales: Este tipo de termómetro posee sensores electrónicos por medio de los que se mide la temperatura, información que es luego convertida por micro-chips a información legible (números y letras), que son observables mediante una pantalla que sirve de visualizador de los datos concernientes a la temperatura medida.



Figura 6: Termómetro de tubo de vidrio y digital, elEconomista, <https://www.eleconomistaamerica.cl/actualidad-eAm-chile/noticias/10444510/03/20/Que-hacer-si-se-rompe-el-termometro-en-casa.html>

2.3.6 Frecuencia Cardíaca

Al hablar de frecuencia cardíaca se debe referir primero al funcionamiento del corazón que es el organismo encargado de impulsar la sangre a través del sistema circulatorio; este proceso consiste en una sucesión de fenómenos químicos, eléctricos, mecánicos y sonoros, conocido como ciclo cardíaco.

Fenómeno Químico

Tanto el sistema nervioso como los músculos del cuerpo se excitan debido a intercambios químicos de iones ocurriendo en cada una de las células que lo componen. En las fibras cardiacas iones de calcio y sodio se intercambian entre las fibras del miocardio y la sangre de los vasos, provocando así una excitación que produce los movimientos mecánicos del corazón.

Fenómeno eléctrico

Debido a este intercambio de iones en las fibras cardiacas del miocardio de la aurícula derecha, origina un impulso eléctrico que se propaga por el resto del sistema automático: el nódulo aurículo-ventricular y el haz de His, con sus ramas izquierda y derecha y las fibras de Purkinje.

Fenómeno mecánico

En respuesta a este impulso eléctrico el corazón se contrae reduciendo sus cavidades, esta reducción se denomina sístole. Cuando el corazón recobra su forma, sus cavidades se dilatan y se relaja, se conoce como diástole. Durante estas las aurículas y los ventrículos se llenan de sangre, mientras que, durante las sístoles, la sangre es arrojada de la cavidad. Esta serie de movimientos permite al corazón actuar como una bomba formando el ciclo cardiaco.

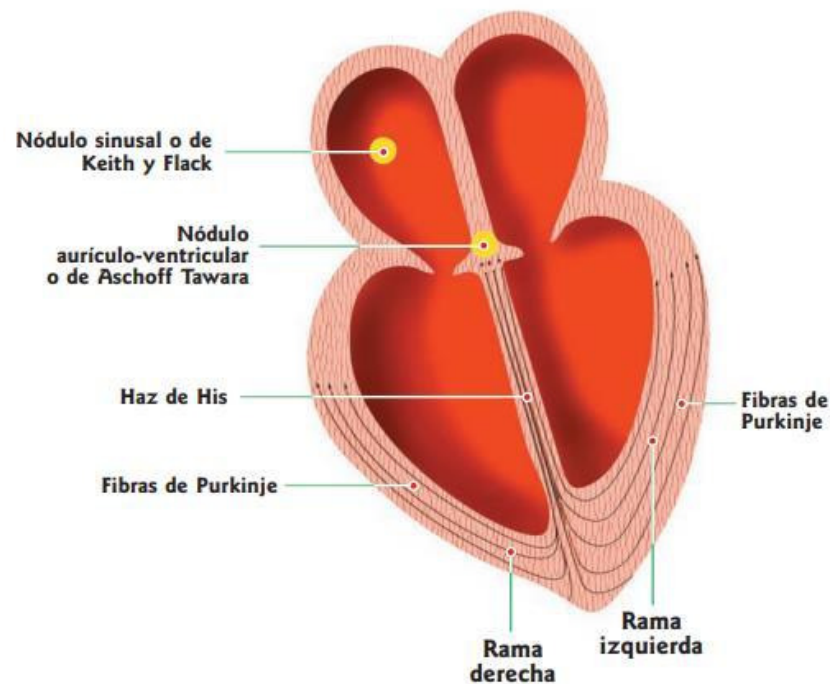


Figura 7: Partes del Corazón, Fenómeno Eléctrico, Anatomía y fisiología del cuerpo humano.

La frecuencia cardiaca consiste en el número de ciclos cardíacos que ocurren en un minuto. En condiciones normales, oscila entre 60 y 90 ciclos cardíacos por minuto en adultos. Estos valores cambian a lo largo de la vida ya que disminuyen con la edad.

Para un recién nacido esta frecuencia se aproxima a los 140 por minuto, en el caso de un niño llega a los 90 ciclos por minuto, y para un adulto ronda los 75 ciclos por minuto. (Vay, 2019)

2.3.7 Electrocardiografo

Este instrumento permite obtener una representación gráfica de la actividad eléctrica del corazón en función del tiempo, conocido como electrocardiograma, además se puede obtener la duración del ciclo cardiaco.

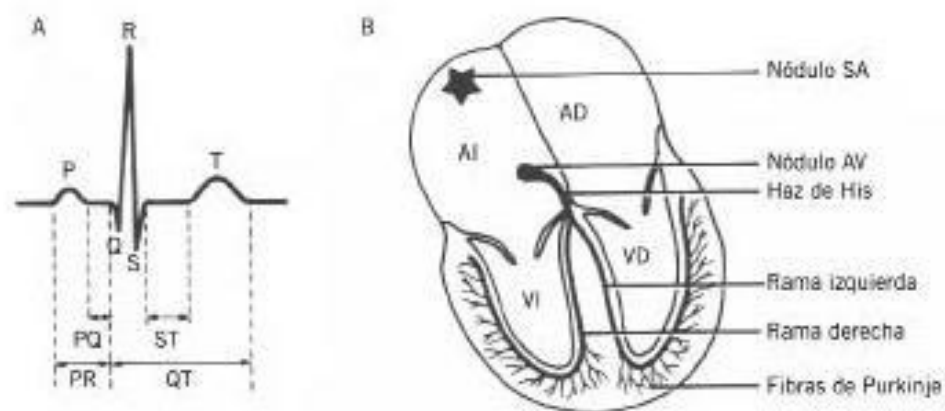


Figura 8: Ciclo cardiaco normal (A) y sistema de conducción del corazón (B), *Pediatr Integral* 2012; XVI(9): 715-722, <https://www.pediatriaintegral.es/numeros-antteriores/publicacion-2012-11/lectura-del-ecg/>

El nódulo sinusal en la aurícula derecha es el marcapasos del corazón, genera el impulso eléctrico, este despolariza las aurículas, primero la aurícula derecha (AD) y luego la aurícula izquierda (AI), produciendo la onda P.

Al llegar al nódulo auriculoventricular (AV), debido a que tiene la velocidad de conducción más lenta de todo el corazón, se produce el intervalo PR (enlentecimiento del nódulo AV). Una vez alcanza el haz de His por ambas ramas, izquierda y derecha, que tienen una velocidad de conducción muy rápida, llega a los ventrículos a través de las fibras de Purkinje, generando la perturbación QRS.

La repolarización ventricular produce la onda T, mientras que la repolarización auricular no suele ser visible. Finalmente, un ciclo cardiaco está representado en el electrocardiograma por una serie de ondas P, complejo QRS y onda T, con dos intervalos (PR y QT) y segmentos (PQ, ST y TP). (Matamala, 2012)



Figura 9: Electrocardiografo, MedicalTec, <https://medicaltec.cl/shop/electrocardiografos>

2.3.8 Tensión Arterial

La tensión arterial se define como la fuerza que ejerce la sangre sobre las paredes de las arterias, ésta es esencial para la circulación de sangre en los vasos sanguíneos y a través de ésta, llevar el oxígeno y los nutrientes a todos los tejidos del organismo, para así mantener sus actividades de manera correcta.

Existen dos tipos de tensión arterial, la presión sistólica, que corresponde a la más alta y ocurre cuando el corazón bombea hacia las arterias y, por otro lado, la presión diastólica es más baja y ocurre entre un latido y otro. Ambos valores se utilizan para medir la tensión arterial, y tienen mucha importancia a la hora de evaluar el estado de salud general, ya que el caso de estar por encima de lo normal (hipertensión) puede significar un importante riesgo de sufrir enfermedades graves como enfermedades cardíacas, infarto cardíaco, ictus o insuficiencia renal. Cuando es baja (hipotensión) puede ocasionar estados de confusión, mareos, vértigo, desmayos, debilidad o somnolencia.

Según los valores de tensión arterial de ambos tipos, describiendo primero la tensión sistólica seguido de la diastólica, se puede clasificar al paciente del siguiente modo:

- Normal: se determina la normalidad con valores que oscilan entre los 90/60 y 130/90 mm de mercurio, se pueden encontrar valores de oscilación más precisos según la edad del paciente, sin embargo, estos valores los abarcan en su totalidad.
- Hipotensión o tensión baja: habitualmente se produce cuando se genera una caída de 20 mm de mercurio sobre los valores normales.
- Hipertensión o tensión alta: si se superan los 140/90 mm de mercurio. El envejecimiento a menudo conduce a un aumento de la presión arterial debido al endurecimiento de las paredes arteriales, por lo que es común que los adultos mayores desarrollen presión arterial alta.
- Prehipertensión: clasificación incorporada recientemente, se determina cuando los valores de presión arterial se encuentran entre 130/80 y 140/90 mm Hg. La razón es que los expertos creen que es hora de comenzar a controlar la presión arterial de manera más continua y tomar medidas preventivas que afecten su estilo de vida, especialmente en lo que respecta a la dieta y el ejercicio regular, así como dejar de fumar si es fumador.

Es importante señalar que los valores de presión arterial no se mantienen constantes a lo largo del día, sino que fluctúan en función de diferentes factores. Esta clasificación define la presencia o ausencia de condiciones patológicas. Sin embargo, en algunos casos, la presión arterial puede subir o bajar a tiempo, pero esto no significa necesariamente que haya un problema de presión arterial.

Así, las situaciones estresantes o peligrosas pueden producir un aumento temporal pero significativo debido a un incremento en las secreciones de epinefrina y norepinefrina o a la activación del sistema nervioso parasimpático, lo que puede conducir a un aumento de

la frecuencia cardíaca. En cambio, la deshidratación, el consumo de alcohol, la toma de determinados medicamentos, la hipoglucemia o estados de shock (anafilaxia, infarto, etc.), entre otros motivos, pueden provocar un descenso repentino de la presión arterial. (Sanitas, 2000)

2.3.9 Tensiómetro

El tensiómetro o esfigmomanómetro es un dispositivo que mide la fuerza que ejerce el flujo sanguíneo sobre las paredes de las arterias y detecta dos tipos de presiones: sistólica y diastólica.

Existe una gran variedad de tensiómetros, los más utilizados y fáciles de encontrar son los digitales y los manuales o aneroides.

Tensiómetro manual o anerode: cuentan con un brazalete, una bomba de aire, un estetoscopio y un indicador de medición. Este tipo de dispositivos son utilizados por profesionales de la salud con conocimientos técnicos sobre la medición de la presión arterial.

Tensiómetro digital: permiten monitorear las mediciones de presión arterial sin necesidad de contar con conocimientos específicos, por lo que son más fáciles de utilizar.

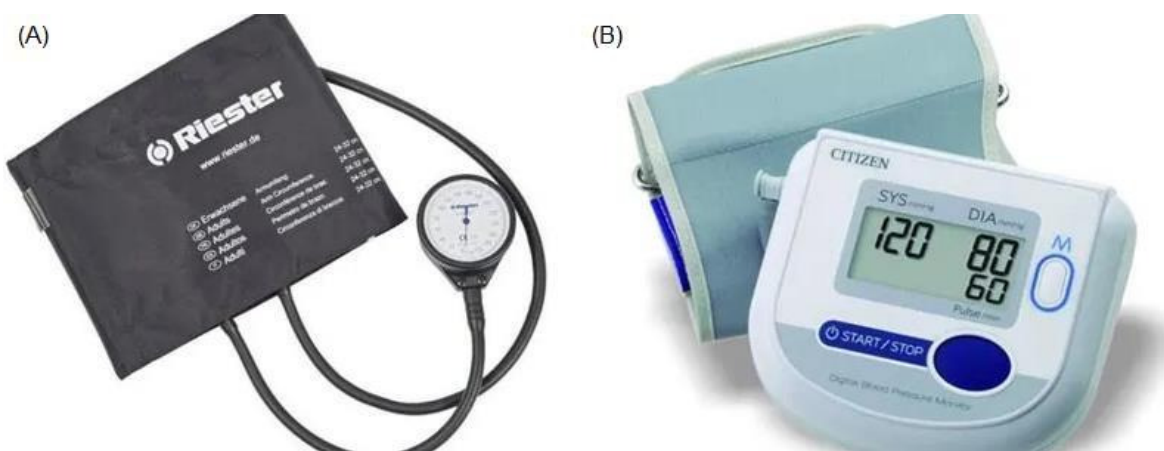


Figura 10: Anerode (A) y tensiómetro digital (B), <https://hiraoka.com.pe/blog/post/tensiometro-que-es-para-que-sirve-y-como-funciona>

Los tensiómetros permiten detectar anomalías en la presión arterial de nuestro cuerpo. La detección temprana de estas irregularidades ayuda a prevenir complicaciones o consecuencias graves para la salud. Además, para personas hipertensas les facilita el seguimiento de la presión sistólica y diastólica desde el hogar.

La mayoría de las mediciones de presión arterial tienen dos números. El número más alto corresponde a la presión sistólica y el más bajo a la diastólica. Si el tensiómetro muestra los números 120 y 80, por ejemplo, la presión arterial correspondería a 120 sobre 80 o más precisamente 120/80 mm Hg. (Hiraoka, 2022)

2.4 Tecnologías a implementar

Aclarados los datos que se recopilarán para el desarrollo del proyecto, deben establecer las herramientas para la obtención y manejo de estos datos, comenzando por los sensores de datos biométricos, que se encargarán de obtener los datos que entrega el cuerpo y transformarlos a señales eléctricas que serán recibidas por una Arduino, un microcontrolador, cuya función es pasar estas señales a eléctricas a lenguaje de máquina para que sea finalmente recibido en la minicomputadora Raspberry pi. Al implementar cada una de estas tecnologías, el proyecto será capaz de realizar IoT al estado de salud del paciente.

2.4.1 Sensores

Corresponde a un dispositivo capaz de detectar magnitudes físicas, químicas, eventos o cambios en su entorno, llamadas variables de instrumentación, y transformarlas en variables eléctricas, para poder enviar esta información a otros componentes electrónicos.

- Las variables de instrumentación pueden ser: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, etc.
- Las variables eléctricas son magnitud de energía como puede ser una resistencia eléctrica como en una RTD (Sensor de temperatura), una capacidad eléctrica como en un sensor de humedad o un sensor capacitivo, una corriente eléctrica como en un fototransistor, etc.

Los sensores se pueden clasificar en función de los datos de salida en:

- Digitales: Los sensores digitales pueden dar una señal digital simple con dos estados, es decir, cambio detectado o cambio no detectado. Lo que es lo mismo: si o no, 1 o 0, activado o desactivado.
- Analógicos: los sensores analógicos muestran más abanico de estados en función de la magnitud física que detectan, haciendo que esta se pueda escalar y obtener el valor real (Solectro, 2021).

Características de los sensores

- Rango de medida: dominio en la magnitud medida en el que puede aplicarse el sensor.
- Precisión: es el error de medida máximo esperado.
- Offset o desviación de cero: valor de la variable de salida cuando la variable de entrada es nula. Si el rango de medida no llega a valores nulos de la variable de entrada, habitualmente se establece otro punto de referencia para definir el offset.
- Linealidad o correlación lineal.

- Sensibilidad de un sensor: suponiendo que es de entrada a salida y la variación de la magnitud de entrada.
- Resolución: mínima variación de la magnitud de entrada que puede detectarse a la salida.
- Rapidez de respuesta: puede ser un tiempo fijo o depender de cuánto varíe la magnitud a medir. Depende de la capacidad del sistema para seguir las variaciones de la magnitud de entrada.
- Derivas: son otras magnitudes, aparte de la medida como magnitud de entrada, que influyen en la variable de salida. Por ejemplo, pueden ser condiciones ambientales, como la humedad, la temperatura u otras como el envejecimiento (oxidación, desgaste, etc.) del sensor.
- Repetitividad: error esperado al repetir varias veces la misma medida (Ginés, 2019).

2.4.2 ESP32

ESP32 un chip de bajo costo y consumo, además de tamaño compacto, de la familia SoC (*System on Chip*), integrado con tecnología de *Wi-Fi* y *Bluetooth*. Emplea un microprocesador Tensilica Xtensa LX6 en sus variantes de simple y doble núcleo e incluye interruptores de antena, balun de radiofrecuencia, amplificador de potencia, amplificador receptor de bajo ruido, filtros, y módulos de administración de energía.

Este chip tiene la habilidad de interactuar con el mundo exterior, por medio de sensores externos al igual que el microcomputador Arduino, y puede ser usada en una amplia variedad de proyectos digitales.

2.4.3 Max30102:

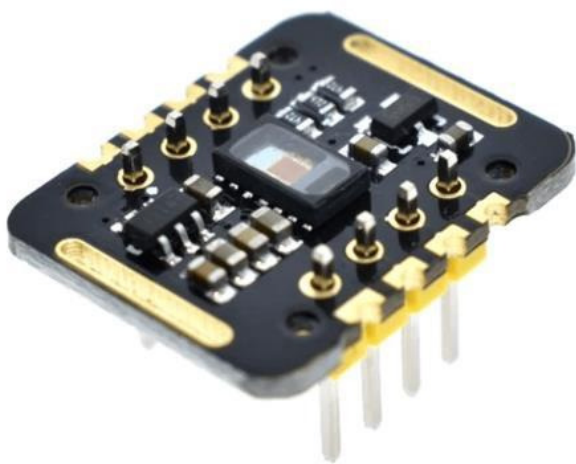


Figura 11: Sensor de pulso y Oxímetro MAX30102, <https://www.mechatronicstore.cl/sensor-de-pulso-y-oximetro-max30102-arduino/>

Este biosensor tiene dos LED: uno rojo y otro IR, además de un fotodetector muy sensible. La idea es iluminar un solo LED cada vez, detectar la cantidad de luz que le devuelve el detector y, en base a esto, medir el nivel de oxígeno en sangre y la frecuencia cardíaca.

Por medio de la librería *SparkFun MAX3010x*, se podrá interactuar con el dispositivo para recopilar datos, pasar por cada una de sus funciones y activar o desactivar el biosensor.

2.4.3.1 Funcionamiento:

Este biosensor, al igual que cualquier otro pulsioxímetro óptico y sensor de frecuencia cardíaca, consta de los LED de alta intensidad rojo e IR, como se mencionó anteriormente, con longitudes de onda de 660 nm y 880 nm, respectivamente, y un fotodetector.

El MAX30102 funciona haciendo brillar ambas luces sobre el dedo o el lóbulo de la oreja (o cualquier lugar donde la piel no sea demasiado gruesa, de modo que ambas luces puedan penetrar fácilmente en el tejido), midiendo la cantidad de luz reflejada mediante un fotodetector. Este método de detección del pulso a través de la luz se denomina Fotopleetismograma.

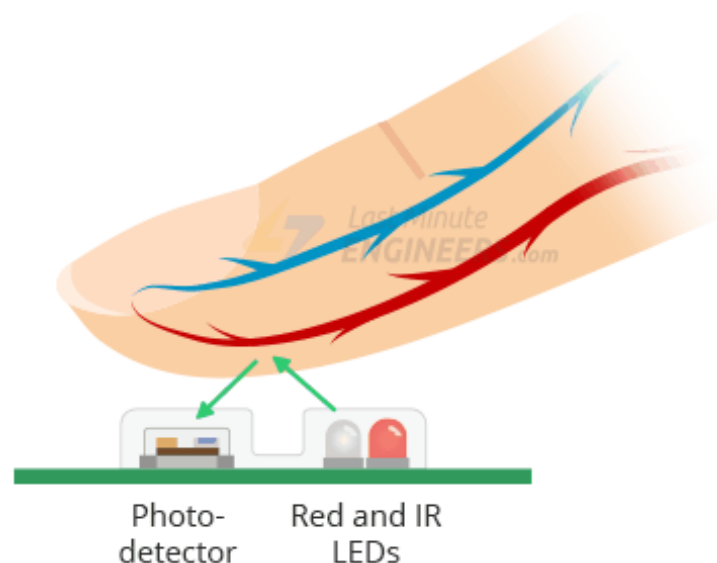


Figura 12: Detección de pulso max30102, <https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/>

Su funcionamiento puede dividirse en dos partes: Medición de la Frecuencia Cardíaca y Pulsioximetría (medición del nivel de oxígeno en la sangre). (Last Minutes Engineers, 2018)

- Medición de la frecuencia cardíaca: La hemoglobina oxigenada (HbO₂) de la sangre arterial tiene la característica de absorber la luz IR. Cuanto más roja es la sangre, más alta es la hemoglobina y más luz IR se absorbe. A medida que la sangre es bombeada a través del dedo con cada latido del corazón, la cantidad de luz reflejada cambia, creando una forma de onda cambiante en la salida del fotodetector. A medida que se sigue emitiendo luz y tomando lecturas del fotodetector, empieza a obtener una lectura del pulso del latido cardíaco (FC).

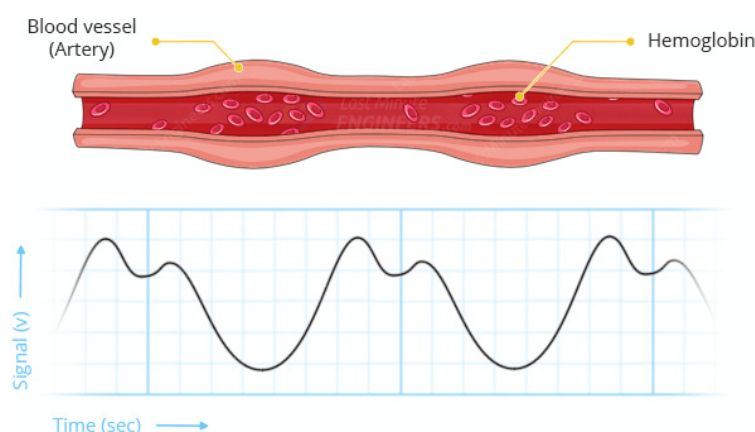


Figura 13: Detección del pulso sensor de frecuencia cardíaca, <https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/>

- Pulsioximetría: La pulsioximetría se basa en el principio de que la cantidad de luz ROJA e IR absorbida varía en función de la cantidad de oxígeno en la sangre. El siguiente gráfico (Figura 14) muestra el espectro de absorción de la hemoglobina oxigenada (HbO₂) y la hemoglobina desoxigenada (Hb). Como puede verse en el gráfico, la sangre desoxigenada absorbe más luz ROJA (660 nm), mientras que la sangre oxigenada absorbe más luz IR (880 nm). Midiendo la proporción de luz IR y ROJA recibida por el fotodetector, se calcula el nivel de oxígeno (SpO₂) en la sangre.

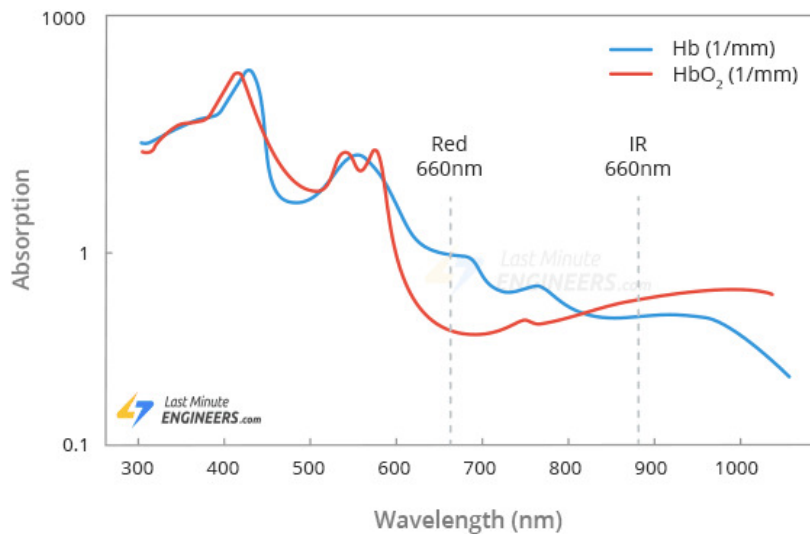


Figura 14: Espectro de absorción de la hb y la hbo2, <https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/>

2.4.4 Bluetooth

El término Bluetooth describe una tecnología de red desarrollada por el grupo de trabajo IEEE 802.15.1 del *Institute of Electrical and Electronics Engineers* estadounidense como estándar industrial para conexiones inalámbricas. La tecnología Bluetooth sirve para la transferencia de voz y datos punto a punto sin conexión u orientada a la conexión entre dos dispositivos digitales diferentes.

En comparación con otras tecnologías de transferencia de datos como USB, LAN o Wi-Fi, Bluetooth está especializada en la transferencia de datos en distancias cortas, así como en el establecimiento de conexiones sencillas y de bajo consumo. Puesto que, en comparación con las demás tecnologías mencionadas, en general solo alcanza velocidades bajas de transferencia de datos, el envío de paquetes grandes puede requerir algo más de tiempo. (Digital Guide IONOS, 2022)

2.4.5 AD8232 Sensor de ECG

Este sensor es una placa económica que se utiliza para medir la actividad eléctrica del corazón. Esta actividad eléctrica puede ser representada como un ECG o Electrocardiograma y la salida como una lectura analógica. Los ECGs pueden ser extremadamente ruidosos, el monitor de frecuencia cardíaca de una sola derivación AD8232 actúa como un op-amp para ayudar a obtener una señal clara de los intervalos PR y QT fácilmente.

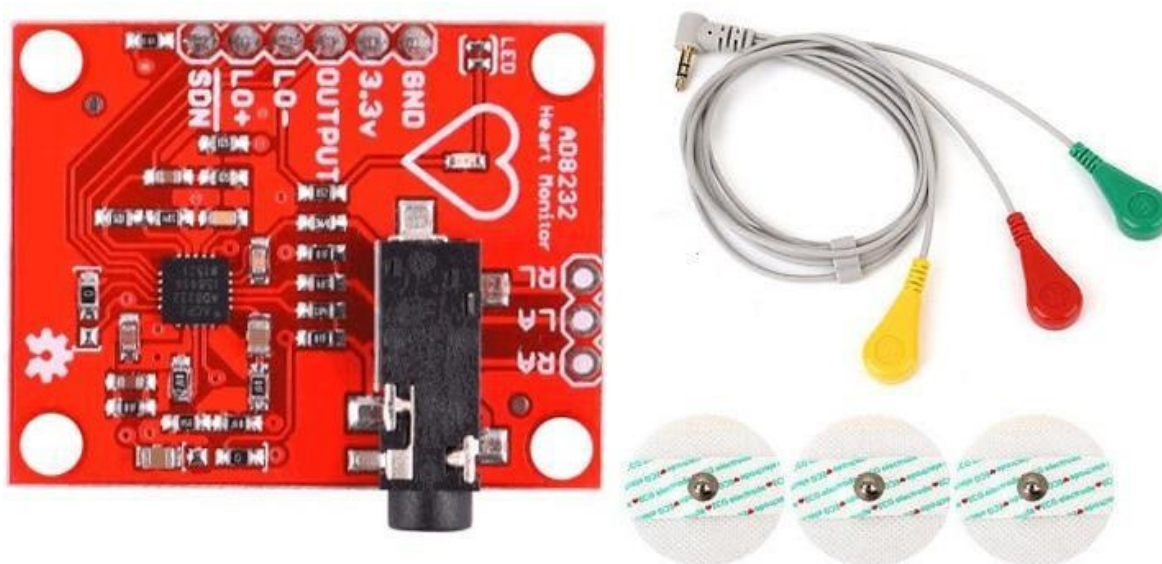


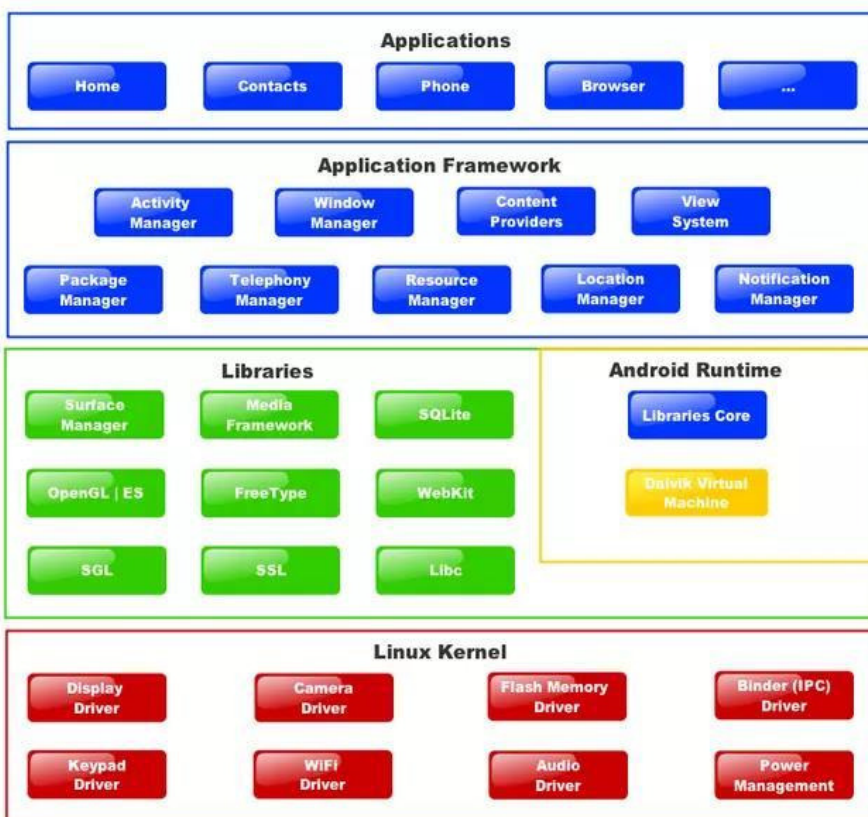
Figura 15: AD8232 ECG Sensor, How to Electronics, <https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/>

El AD8232 es un bloque integrado de acondicionamiento de señal para ECG y otras aplicaciones de medición biopotencial. Está diseñado para extraer, amplificar y filtrar pequeñas señales biopotenciales en presencia de condiciones ruidosas, como las creadas por el movimiento o la colocación remota de electrodos. (How to Electronics, 2022)

2.4.6 Android

Android es un sistema operativo inicialmente pensado para teléfonos móviles, al igual que iOS, Symbian y BlackBerry OS. Lo que lo hace diferente es que está basado en Linux, un núcleo de sistema operativo libre, gratuito y multiplataforma.

El sistema permite programar aplicaciones en una variación de Java llamada Dalvik. El sistema operativo proporciona todas las interfaces necesarias para desarrollar aplicaciones que accedan a las funciones del teléfono (como el GPS, las llamadas, la agenda, etc.) de una forma muy sencilla en un lenguaje de programación muy conocido como es Java.



Esta sencillez, junto a la existencia de herramientas de programación gratuitas, hacen que una de las cosas más importantes de este sistema operativo sea la cantidad de aplicaciones disponibles, que extienden casi sin límites la experiencia del usuario. (Gonzalez, 2011)

Figura 16: Esquema de Arquitectura, Xataka Android, <https://www.xatakandroid.com/sistema-operativo/que-es-android>

2.4.7 Android Studio

Android Studio es el entorno de desarrollo integrado (IDE) oficial que se usa en el desarrollo de apps para Android. Basado en el potente editor de código y las herramientas para desarrolladores de IntelliJ IDEA, Android Studio ofrece aún más funciones que mejoran tu productividad cuando compilas apps para Android, como las siguientes:

- Un sistema de compilación flexible basado en Gradle
- Un emulador rápido y cargado de funciones
- Un entorno unificado donde puedes desarrollar para todos los dispositivos Android

- Aplicación de cambios para insertar cambios de código y recursos a la app en ejecución sin reiniciarla
- Integración con GitHub y plantillas de código para ayudarte a compilar funciones de apps comunes y también importar código de muestra
- Variedad de marcos de trabajo y herramientas de prueba
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones, entre otros
- Compatibilidad con C++ y NDK
- Compatibilidad integrada con Google Cloud Platform, que facilita la integración con Google Cloud Messaging y App Engine

(Android Studio, 2023)

2.4.8 OpenCV

OpenCV (Open Source Computer Vision) es una librería software open-source de visión artificial y machine learning, desarrollada originalmente por Intel. Vio la luz en el año 1999. Escrita originalmente en C/C++, su mejor virtud es que es multiplataforma, se puede ejecutar en diferentes sistemas operativos (Linux, Windows, Mac OS X, Android e iOS). También la podemos utilizar en diferentes lenguajes de programación como Java, Objective C, Python y C#. (Hernández, 2021)

2.4.8.1 Visión Artificial

La visión artificial es un campo de la IA que permite que las computadoras y los sistemas obtengan información significativa de imágenes digitales, videos y otras entradas visuales, y tomen acciones o hagan recomendaciones basadas en esa información.

La visión artificial entrena a las máquinas para realizar estas funciones, pero tiene que hacerlo en mucho menos tiempo con cámaras, datos y algoritmos en lugar de retinas, nervios ópticos y una corteza visual. (IBM, 2019)

2.4.8.2 Reconocimiento de imágenes

Un algoritmo de reconocimiento de imágenes (también conocido como clasificador de imágenes) toma una imagen (o un fragmento de una imagen) como entrada y da como resultado lo que contiene la imagen. En otras palabras, la salida es una etiqueta de clase (por ejemplo, "gato", "perro", "mesa", etc.). Para llegar a esto, hay que entrenar al algoritmo para que aprenda las diferencias entre las distintas clases. Por ejemplo, si quiere encontrar gatos en imágenes, tiene que entrenar un algoritmo de reconocimiento de imágenes con miles de imágenes de gatos y miles de imágenes de fondos que no contengan gatos. (Mallick, 2016)

2.4.9 Firebase

Firebase de Google es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Está disponible para distintas plataformas (iOS, Android y web)

Fue creada en 2011, y pasó a ser parte de Google en 2014, comenzando como una base de datos en tiempo real. Sin embargo, se añadieron más funciones, permitiendo agrupar los SDK de productos de Google con distintos fines, para facilitar su uso.

Su función esencial es hacer más sencilla la creación de tanto aplicaciones webs como móviles y su desarrollo, procurando que el trabajo sea más rápido.

Sus herramientas son variadas y de fácil uso, considerando que su agrupación simplifica las tareas de gestión a una misma plataforma. Las finalidades de las mismas se pueden dividir en cuatro grupos: desarrollo, crecimiento, monetización y análisis. (Mora, 2020)

2.4.9.1 Firebase Realtime Database

Firebase Realtime Database es una base de datos alojada en la nube. Los datos se almacenan en formato JSON y se sincronizan en tiempo real con cada cliente conectado. Cuando compilas apps multiplataforma con nuestros SDK de plataformas de Apple, Android y JavaScript, todos tus clientes comparten una instancia de Realtime Database y reciben actualizaciones automáticamente con los datos más recientes. (Firebase, 2022)

CAPÍTULO 3: PROPUESTA DE SOLUCIÓN

Para resolver esta problemática, se propone crear un sistema capaz de integrar distintos aparatos de monitoreo médico simples, con el objetivo de que los datos obtenidos se puedan almacenar y enviar a personal capacitado para realizar revisiones periódicas remotas, además de generar alertas en el caso de encontrar anomalías en dichas mediciones. Con ello, se espera crear una hoja de vida constante del paciente, facilitando la entrega y consulta de antecedentes de salud, haciendo más expedita la respuesta médica en casos de peligro.

El sistema se orienta especialmente a pacientes cuyos signos vitales deban ser monitoreados de manera periódica, pero permitiendo una mayor movilidad y comodidad a la hora de ser necesario para una evaluación médica, ya sea de urgencia o de rutina.

El sistema permitirá al personal de salud disponer, de información sobre el estado y evolución de sus pacientes desde cualquier lugar con acceso a internet, facilitando las capacidades de tele atención y telemedicina; ayudando además a mejorar los tiempos de respuesta ante situaciones de urgencia, ya que se dispondría de información actualizada sobre la evolución de salud de los pacientes, reduciendo además la necesidad de desplazamiento de éstos a centros de salud.

Con esta solución se buscan los siguientes objetivos a cumplir:

- Capacidad de realizar múltiples mediciones de parámetros médicos simples por medio de sensores biométricos como termómetro, sensor de pulso, oxímetro, tensiómetro, etc.
- Generar un sistema que permita alertar a familiares y médicos en caso de encontrar peligro en alguna variable medida
- Capacidad para almacenar información médica a nivel local y distribuido, y realizar reportes periódicos del estado de salud del paciente y su evolución
- Dar la posibilidad de admitir más tipos de mediciones provenientes de otros dispositivos.

Para cumplir con los objetivos se manejaron distintos sistemas que podrían ser capaces de llegar a una solución, sin embargo, los siguientes fueron los que tuvieron un mayor desarrollo.

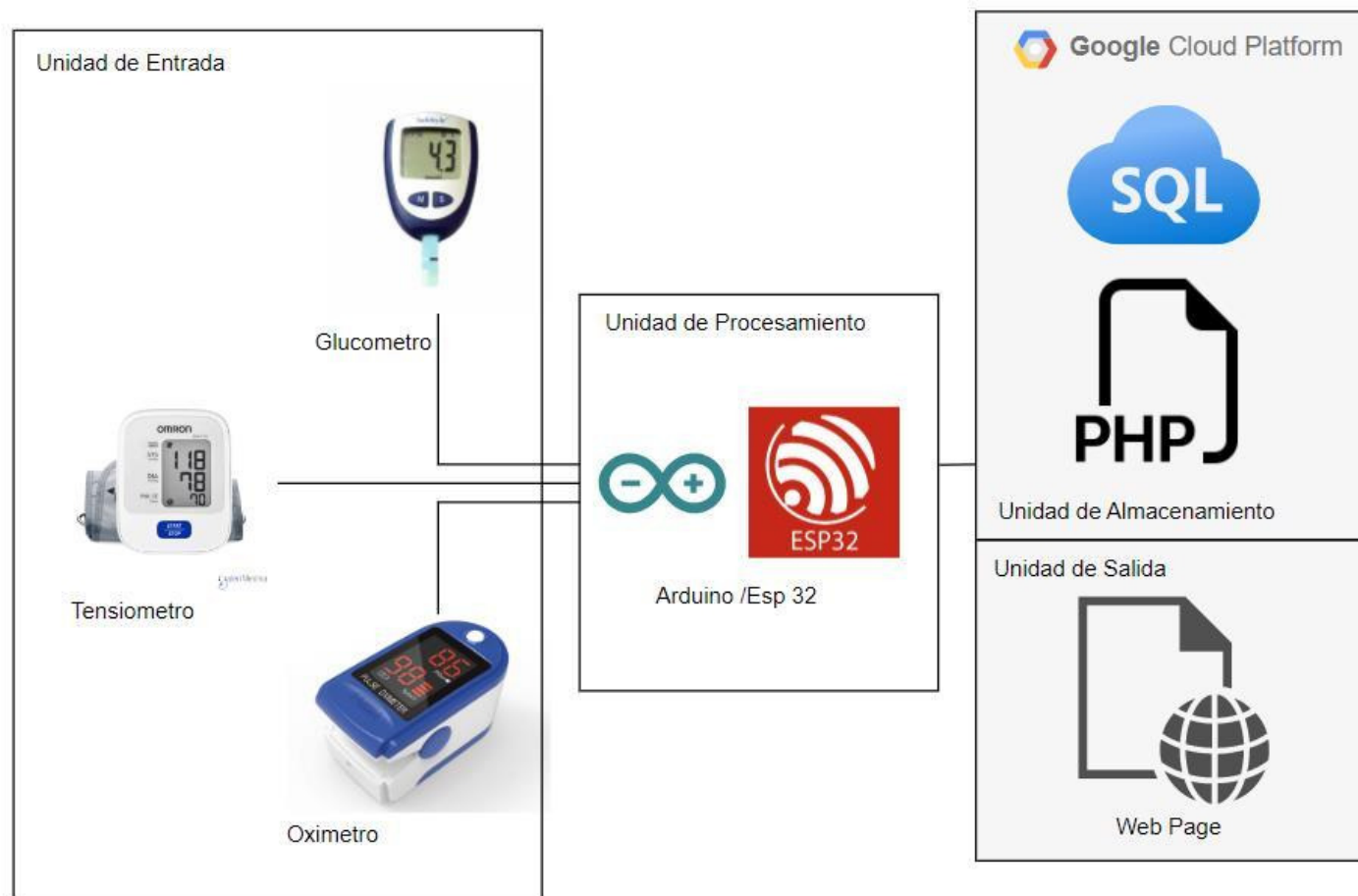


Figura 17: Diagrama de Hardware de solución Inicial, Elaboración Propia

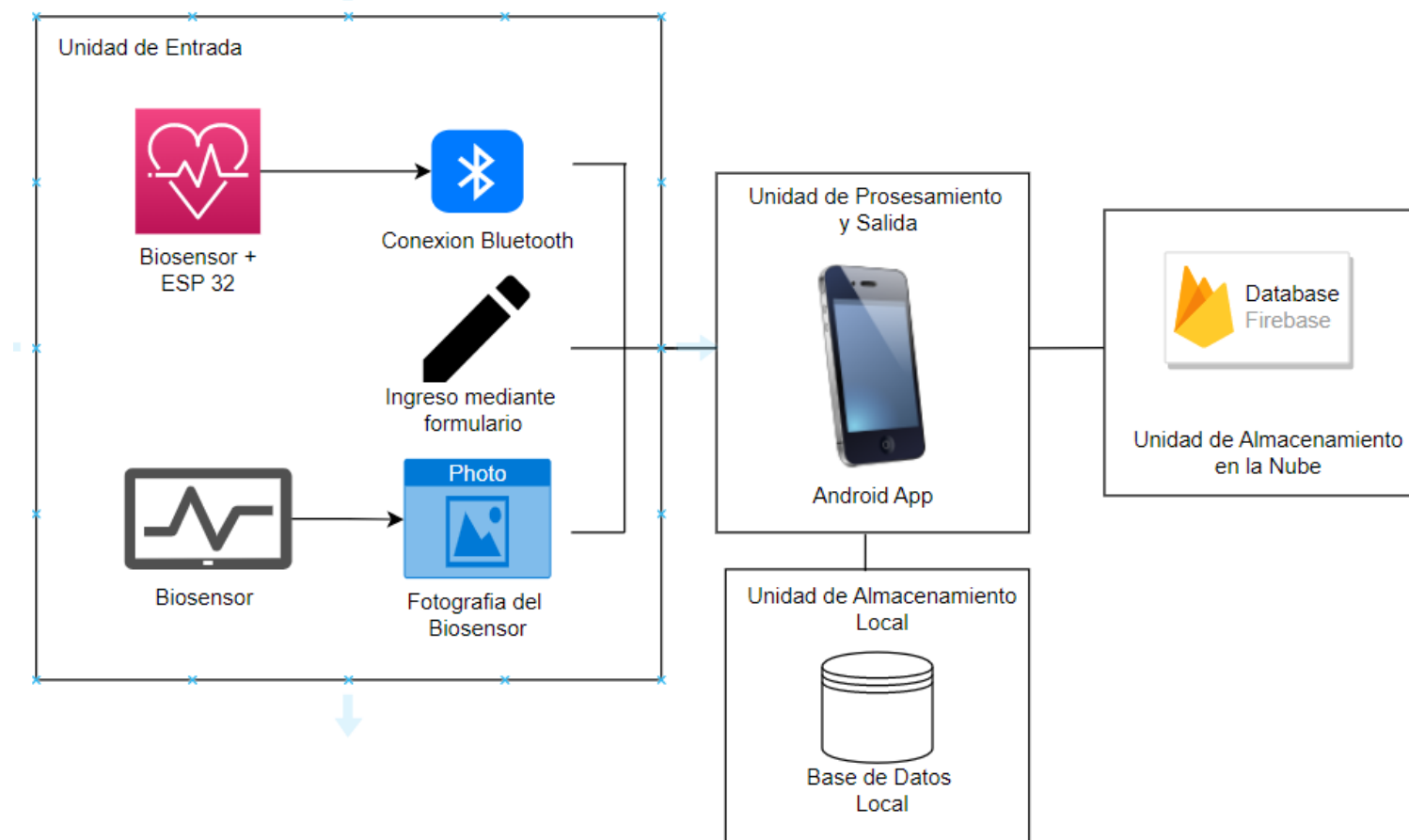


Figura 18: Diagrama de Hardware de solución, Elaboración Propia

El diagrama de la Figura 17 muestra una primera solución, en la cual se utilizarían distintos biosensores del mercado a los cuales se les agregaría un microcomputador Arduino para la obtención directa de los datos que otorgan estos, mediante un microcomputador ESP32 conectada al Arduino se transmitiría los datos por medio de wifi a la base de datos SQL Cloud. Para el acceso a estos datos se utilizaría una página web.

En la Figura 18 se muestra el diagrama de la solución realizada, la que utiliza parte de lo desarrollado en la primera solución, a la cual se le agrega una aplicación para dispositivos con Android, otorgando con ello más flexibilidad al sistema y al usuario para capturar la información esperada.

Para este sistema se tiene un biosensor conectado a una placa ESP32 para conectar a la App de Android mediante Bluetooth, por otro lado, mediante un formulario se puede ingresar la información de manera directa o finalmente la última manera de agregar datos sería por medio de una foto de un biosensor del mercado, la cual se analizaría en la app mediante procesamiento y reconocimiento de imágenes.

La App de Android tendría su propia base de datos local mediante la cual se pueden crear usuarios, lo cuales posteriormente se pueden enlazar con la nube para ya sea subir sus datos o recuperar los que se hayan tomado en otro dispositivo.

3.1 Componentes para crear el sistema:

Para desarrollar la solución, el primer sistema de entrada necesitará una serie de biosensores diseñados para sistemas de Arduino que se encuentran disponibles en el mercado nacional, y una placa ESP32 para recibir la información de dichos sensores. Estos datos serán enviados por bluetooth hacia la aplicación de Android para finalmente almacenarlos a una base de datos local junto con el resto de la información del paciente, donde puede ser enviada a la base de datos en la nube si el paciente o asistente así lo decide.

Las siguientes entradas son directamente en la aplicación por lo que mediante Android Studio se desarrollará un sistema de formularios que sea capaz de recibir la información, por otro lado, utilizando las librerías de OpenCV se realizará un sistema de reconocimiento de imágenes para poder analizar fotos de biosensores y añadir la medición del usuario por esta alternativa, si así lo desea.

Se enlistarán los elementos con los que el proyecto se intentará llevar a cabo:

- ESP32: La función de este chip será recibir y analizar las señales del biosensor, realizar la conexión bluetooth con la aplicación, para mediante esta conexión enviar los datos que se requieran.

- Max30102: Biosensor que se utiliza como monitor de pulso y oxígeno en la sangre.
- ECG(ADB232): Dispositivo que permite realizar un ECG en la aplicación por medio de la ESP32.
- *Protoboard*: Base utilizada para construir prototipos semipermanentes de circuitos electrónicos.
- Pantalla LCD 16x2: Pantalla que puede mostrar 16 caracteres por línea y hay 2 líneas de este tipo. Se utilizará para mostrar información de la ESP32 al usuario.
- Celular con sistema operativo Android: Se utilizará para que el usuario pueda interactuar con el sistema mediante la aplicación.
- Cable de puente Dupont: Cables que permiten realizar conexiones eléctricas entre las diferentes piezas del proyecto y la protoboard, para así permitir su comunicación.

Finalmente, para realizar el código se debe utilizar el lenguaje C++ e ingresarlo a la ESP32 por medio del programa Arduino IDE. Para realizar la conexión a la base de datos se utilizará Firebase, una de las herramientas de Google, que permite mantener una base de datos en tiempo real sincronizada con cada cliente conectado, por último, para realizar la aplicación se utilizará el lenguaje Kotlin y en ambiente de Android Studio Dolphin.

CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN

4.1 Análisis de Requerimientos

Según los objetivos presentados anteriormente, se debe realizar un análisis de los requerimientos funcionales y no funcionales que la aplicación y el sistema deben cumplir.

A continuación, se presentarán los principales:

4.1.1 Requerimientos Funcionales

1. La aplicación debe autenticar a cada usuario antes de tener acceso a los datos y capacidades del sistema.
2. El sistema debe permitir el ingreso de los usuarios mediante un usuario y una contraseña.
3. El Ingreso de datos debe ser adaptable y ofrecer diferentes opciones para el usuario.
4. Datos médicos externos generales del paciente pueden ser ingresados de manera manual.
5. Mediante Bluetooth, la aplicación debe permitir conectarse al biosensor, siempre que este esté cerca del área de servicio.
6. El sistema debe ser capaz de monitorear en tiempo real una la variable médica a través de sensores.
7. Los dispositivos móviles deben cumplir con las características mínimas para la instalación y manejo de la aplicación.
8. El sistema permitirá al usuario llevar un registro de las mediciones que haya ingresado.
9. El sistema permitirá al usuario editar y eliminar las mediciones que haya ingresado.
10. El usuario podrá modificar sus datos personales en su perfil.
11. El sistema tendrá perfiles distintos dependiendo del tipo de usuario que esté utilizando la aplicación.
12. El usuario deberá poder compartir su información con los profesionales de la salud que requiera.
13. Los profesionales de la salud podrán ver a los datos de todos los usuarios que tienen enlazados.
14. El usuario debe poder mantener su información y mediciones de perfil a través de distintos dispositivos.

4.1.2 Requerimientos no Funcionales

1. La aplicación tener una interfaz intuitiva y de fácil manejo para los usuarios.
2. Añadir datos al sistema debe ser rápido y portátil.
3. Las vistas de la aplicación tener un diseño de forma que permitan agilizar el tiempo de ingreso.
4. Los colores de la interfaz gráfica deberán tener coherencia en la aplicación.

4.2 Descripción de Casos de uso

A continuación, se presentarán los casos de uso propuestos, estos surgieron de los requerimientos del usuario y el sistema, con el objetivo de abarcar todas las necesidades de estos, tener una aplicación móvil amigable y con una eficaz respuesta a los requerimientos funcionales. Las siguientes tablas describen los principales casos de uso.

Nombre de caso de uso	<i>Login</i>
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	1-2
Precondiciones	El usuario debe tener una cuenta registrada previamente
Postcondición	El usuario debe ingresar a su <i>Dashboard</i> con su información correspondiente
Descripción	El usuario introducirá su nombre de usuario y su contraseña para iniciar sesión en la aplicación
Flujo	- El usuario introduce su email y contraseña

Tabla 1: Descripción caso de uso *login*, Elaboración Propia

Nombre de caso de uso	<i>Conexión Bluetooth</i>
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	5 - 6
Precondiciones	El usuario debe ingresar a su sesión y tener un dispositivo compatible
Postcondición	El usuario debe poder observar un mensaje de confirmación y los datos del dispositivo conectado
Descripción	El usuario podrá seleccionar en la vista el biosensor al que desee conectarse.
Flujo	- El usuario entrará al menú lateral y escogerá la opción de dispositivo <i>bluetooth</i> - Escoge el dispositivo correspondiente para realizar la medición

Tabla 2: Descripción caso de uso *bluetooth*, Elaboración Propia

Nombre de caso de uso	Registrar datos manual
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	3
Precondiciones	El usuario debe ingresar a su sesión
Postcondición	El usuario debe poder observar una nueva entrada en su lista de mediciones.
Descripción	El usuario podrá registrar mediciones que haya realizado en la aplicación de mediante un formulario
Flujo	- El usuario debe realizar el <i>login</i> y seleccionar la opción

Tabla 3: Descripción caso de uso registró mediciones, Elaboración Propia

Nombre de caso de uso	Registrar datos fotografía
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	3
Precondiciones	El usuario debe ingresar a su sesión
Postcondición	El usuario debe poder observar una nueva entrada en su lista de mediciones, además de la foto analizada por el sistema.
Descripción	El usuario podrá registrar mediciones que haya realizado en la aplicación de mediante una foto del resultado entregado por un biosensor
Flujo	<ul style="list-style-type: none"> - El usuario debe realizar el <i>login</i> y seleccionar la opción - Escoger el tipo de dispositivo al que se le sacará una foto. - Escoger si usar una foto de su galería o sacarla en el momento. - Ajustar la foto para poder ver los valores del biosensor - Confirmar los valores recopilados.

Tabla 4: Descripción caso de uso registrar datos, Elaboración Propia

Nombre de caso de uso	Registrar datos biosensor
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	3
Precondiciones	El usuario debe ingresar a su sesión
Postcondición	El usuario debe poder observar una nueva entrada en su lista de mediciones, además de los datos que entrega el biosensor.
Descripción	El usuario podrá conectarse mediante bluetooth con un biosensor compatible y registrar datos del mismo.
Flujo	<ul style="list-style-type: none"> - El usuario entrará al menú lateral y escogerá la opción de dispositivo bluetooth - Escoge el dispositivo correspondiente para realizar la medición - Iniciar la medición. - Confirmar los valores recopilados.

Tabla 5: Descripción caso de uso datos biosensor, Elaboración Propia

Nombre de caso de uso	Modificar datos usuarios
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	4 - 10
Precondiciones	El usuario debe ingresar a su sesión
Postcondición	El usuario debe poder observar los cambios que realizó en su perfil.
Descripción	El usuario podrá modificar los datos de su perfil, agregar una foto de perfil o agregar datos médicos externos como la edad, peso y altura.
Flujo	- El usuario entrará al menú lateral y escoger la opción de configuración

Tabla 6: Descripción caso de uso modificar usuario, Elaboración Propia

Nombre de caso de uso	Modificar datos mediciones
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	9
Precondiciones	El usuario debe ingresar a su sesión e ingresar alguna medición
Postcondición	El usuario debe poder observar los cambios que realizó en la medición o la desaparición de este en caso que lo eliminara.
Descripción	El usuario podrá modificar o eliminar los datos de las mediciones que se haya tomado.
Flujo	- El usuario entrará al menú lateral y escoger la opción de ver todos los registros - Escoge y seleccionar la medición a modificar

Tabla 7: Descripción caso de uso modificar mediciones, Elaboración Propia

Nombre de caso de uso	<i>Dashboard</i>
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	11
Precondiciones	El usuario debe ingresar a su sesión
Postcondición	El usuario debe tener acceso a todas las funcionalidades de la aplicación y a su perfil.
Descripción	El usuario podrá ingresar a su <i>dashboard</i> que cambiará en función al tipo de perfil, dándole acceso a las herramientas del sistema.
Flujo	- El usuario deberá realizar <i>login</i>

Tabla 8: Descripción caso de uso *dashboard*, Elaboración Propia

Nombre de caso de uso	Afiliar perfiles
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	12 - 13
Precondiciones	El usuario debe ingresar a su sesión y conocer el rut a afiliar

Postcondición	El usuario debe poder observar el nombre y tipo de usuario del perfil a afiliar, además de un mensaje de confirmación en caso de que confirme la acción.
Descripción	El usuario podrá compartir su información con otros mediante la afiliación, que requiere el rut de la persona a afiliar
Flujo	<ul style="list-style-type: none"> - El usuario entrará al menú lateral y escogerá la opción de Afiliar usuario - Ingresar el rut del usuario a ingresar

Tabla 9: Descripción caso de uso afiliar, Elaboración Propia

Nombre de caso de uso	Ver datos usuarios (Pacientes)
Actor	Usuario: Paciente
Referencias	8
Precondiciones	El usuario debe ingresar a su sesión y tener alguna medición
Postcondición	El usuario debe poder observar todas sus mediciones y tener acceso a las modificaciones o eliminación, desde esta lista.
Descripción	El usuario podrá ver todas sus mediciones.
Flujo	<ul style="list-style-type: none"> - El usuario deberá entrar al menú lateral y escoger la opción de ver todos los registros

Tabla 10: Descripción caso de uso ver datos, Elaboración Propia

Nombre de caso de uso	Ver datos usuarios
Actor	Usuario: Asistente, Doctor
Referencias	8 - 13
Precondiciones	El usuario debe ingresar a su sesión y tener algún usuario afiliado
Postcondición	El usuario debe poder observar todas sus mediciones del paciente seleccionado.
Descripción	El usuario podrá seleccionar un usuario afiliado y ver todas sus mediciones.
Flujo	<ul style="list-style-type: none"> - El usuario deberá escoger al paciente a revisar - Entrar al menú lateral y escoger la opción de ver todos los registros

Tabla 11: Descripción caso de uso ver datos, Elaboración Propia

Nombre de caso de uso	Base de datos en la nube
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	14
Precondiciones	El usuario debe ingresar a su sesión
Postcondición	El usuario debe tener sus datos en la nube para poder acceder a ellos mediante otros dispositivos
Descripción	El usuario podrá mantener su perfil y sus datos de dispositivo en dispositivo.
Flujo	<ul style="list-style-type: none"> - El usuario deberá realizar <i>login</i> desde otro dispositivo

Tabla 12: Descripción caso de uso base de datos nube, Elaboración Propia

Nombre de caso de uso	Registrar Usuario
Actor	Usuario: Paciente, Asistente, Doctor
Referencias	1-2
Precondiciones	El usuario debe tener la aplicación
Postcondición	El usuario debe poder ingresar a su cuenta mediante <i>login</i>
Descripción	El usuario podrá registrar su perfil para que pueda hacer uso del sistema y la aplicación
Flujo	<ul style="list-style-type: none"> - El usuario selecciona la opción de registro - Llena el formulario con su tipo de usuario y datos personales - La aplicación realiza una validación de los datos y se crea un nuevo registro en la BD

Tabla 13: Descripción caso de uso registrar usuario, Elaboración Propia

4.3 Diagrama de Casos de uso

Los diagramas de caso de uso modelan las distintas funciones que realiza el sistema, así como la interacción del usuario con el sistema. Los diagramas de casos de uso mostrados a continuación modelan el comportamiento del sistema propuesto, describiendo la interacción del usuario con el sistema.

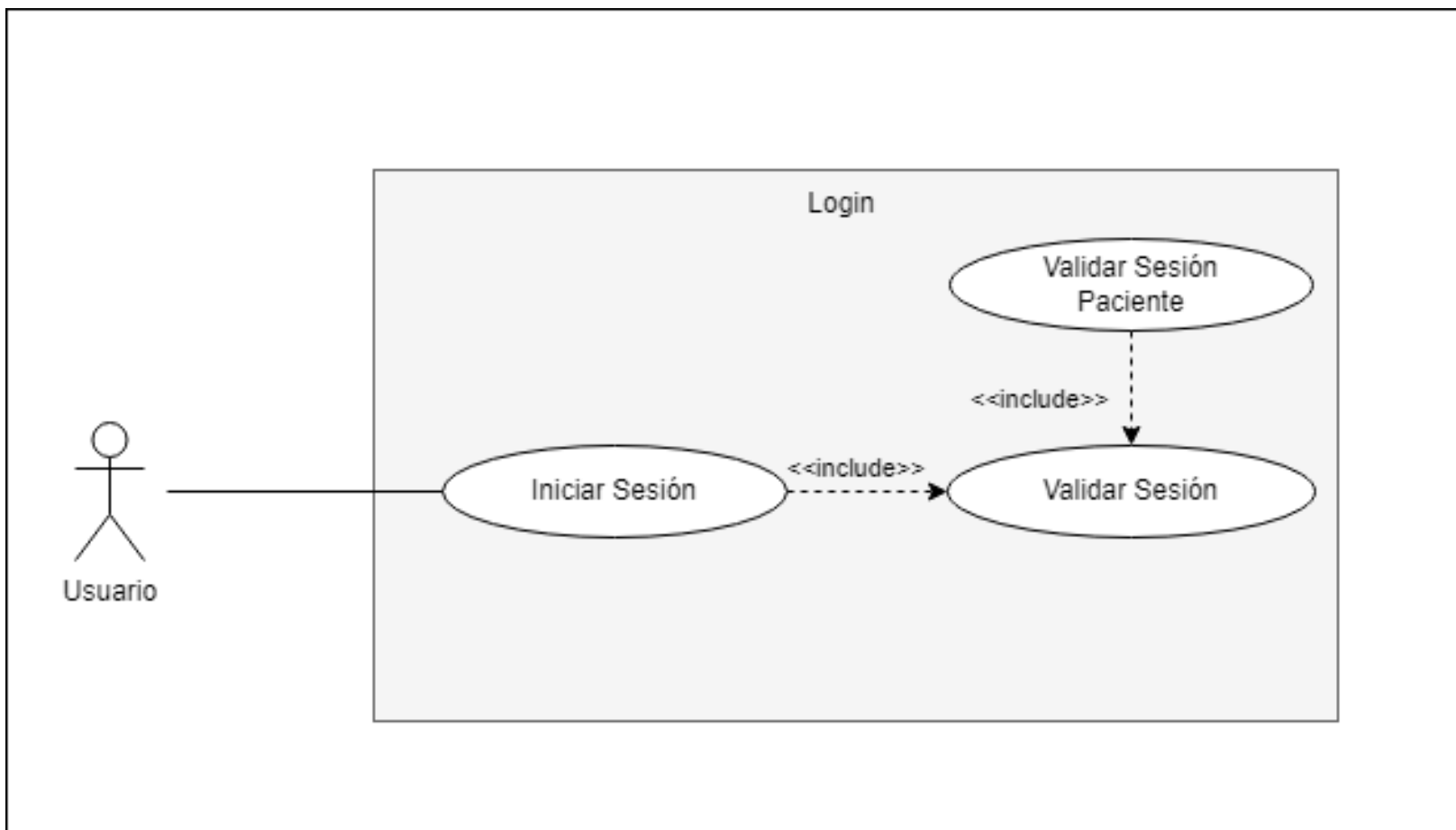


Figura 19: Diagrama caso de uso Login, Elaboración Propia

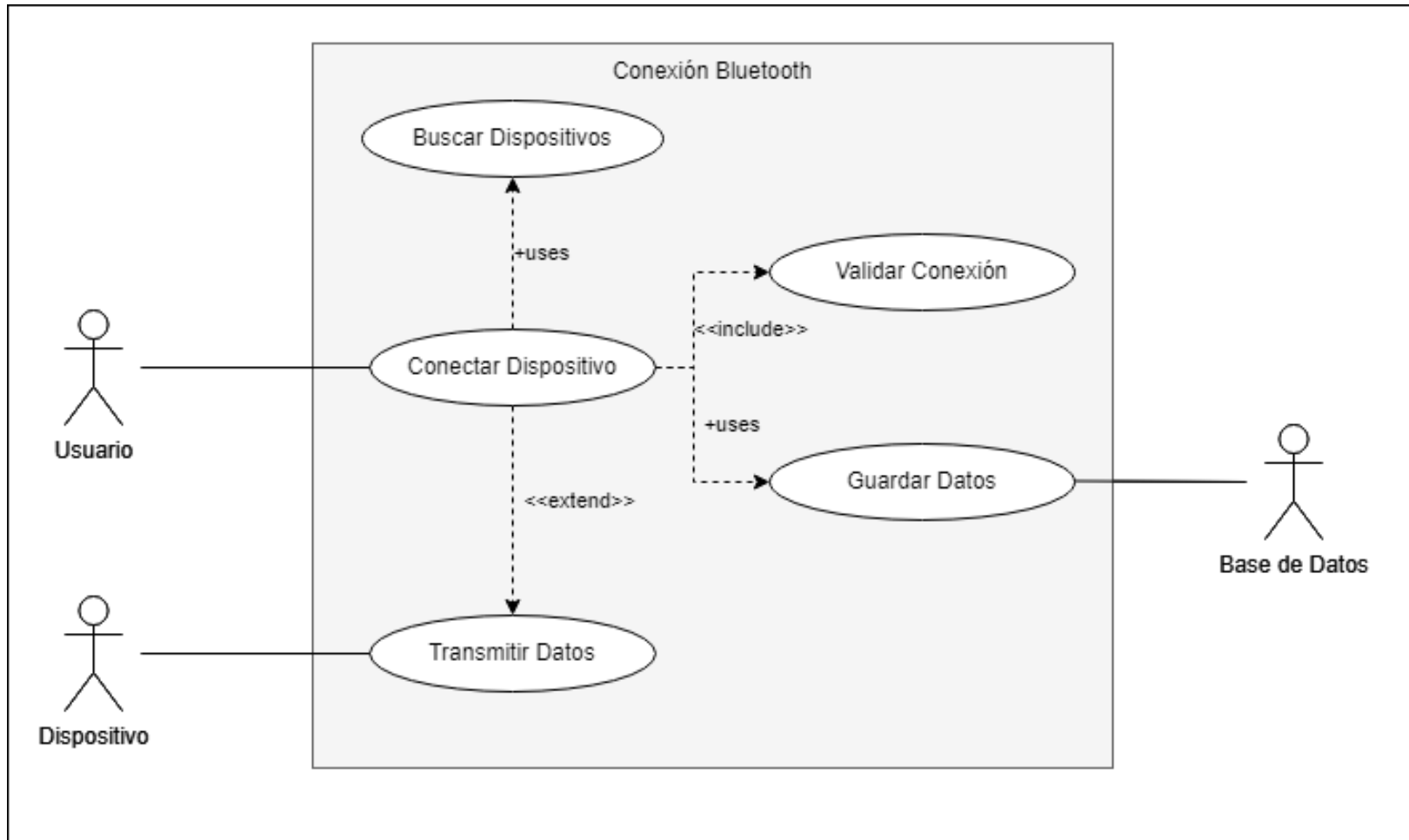


Figura 20: Diagrama caso de uso Conexión BT, Elaboración Propia

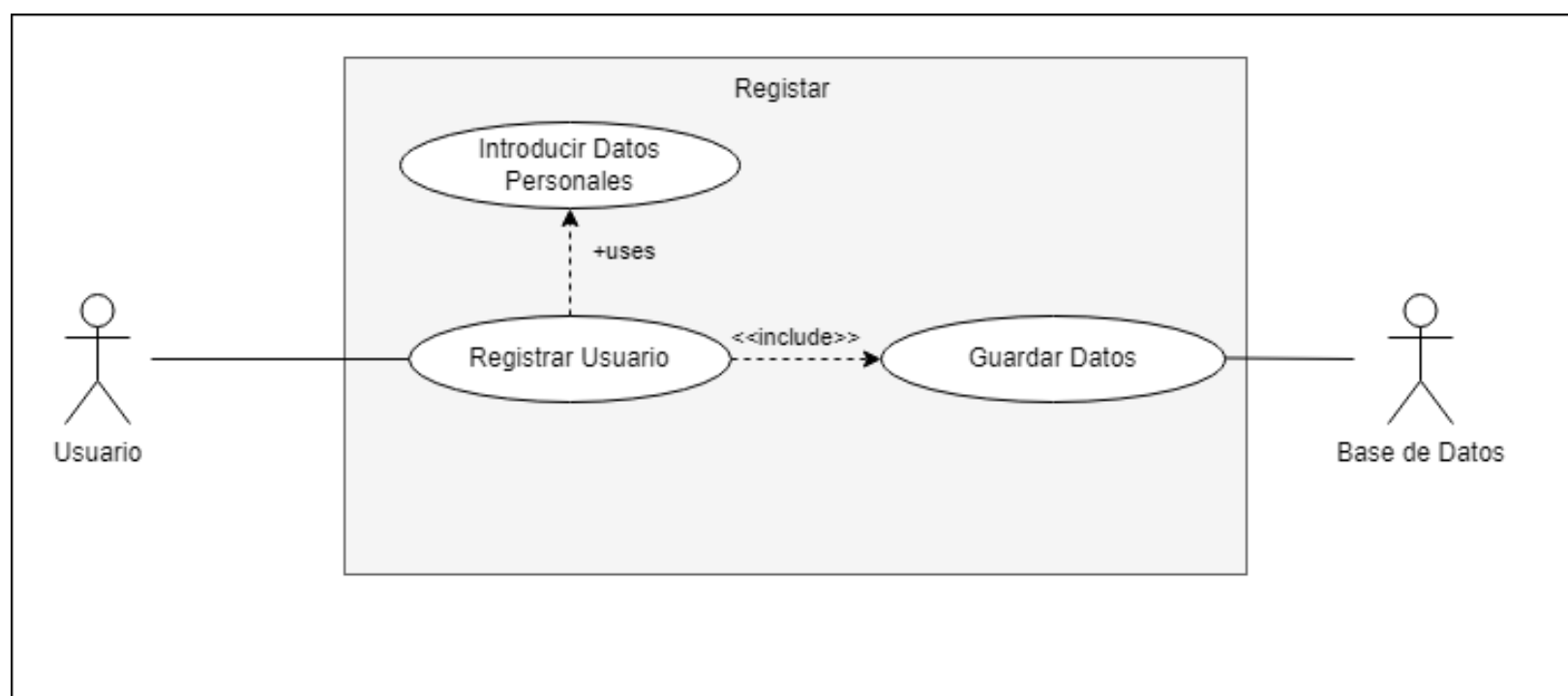


Figura 21: Diagrama caso de uso Registrar usuarios, Elaboración Propia

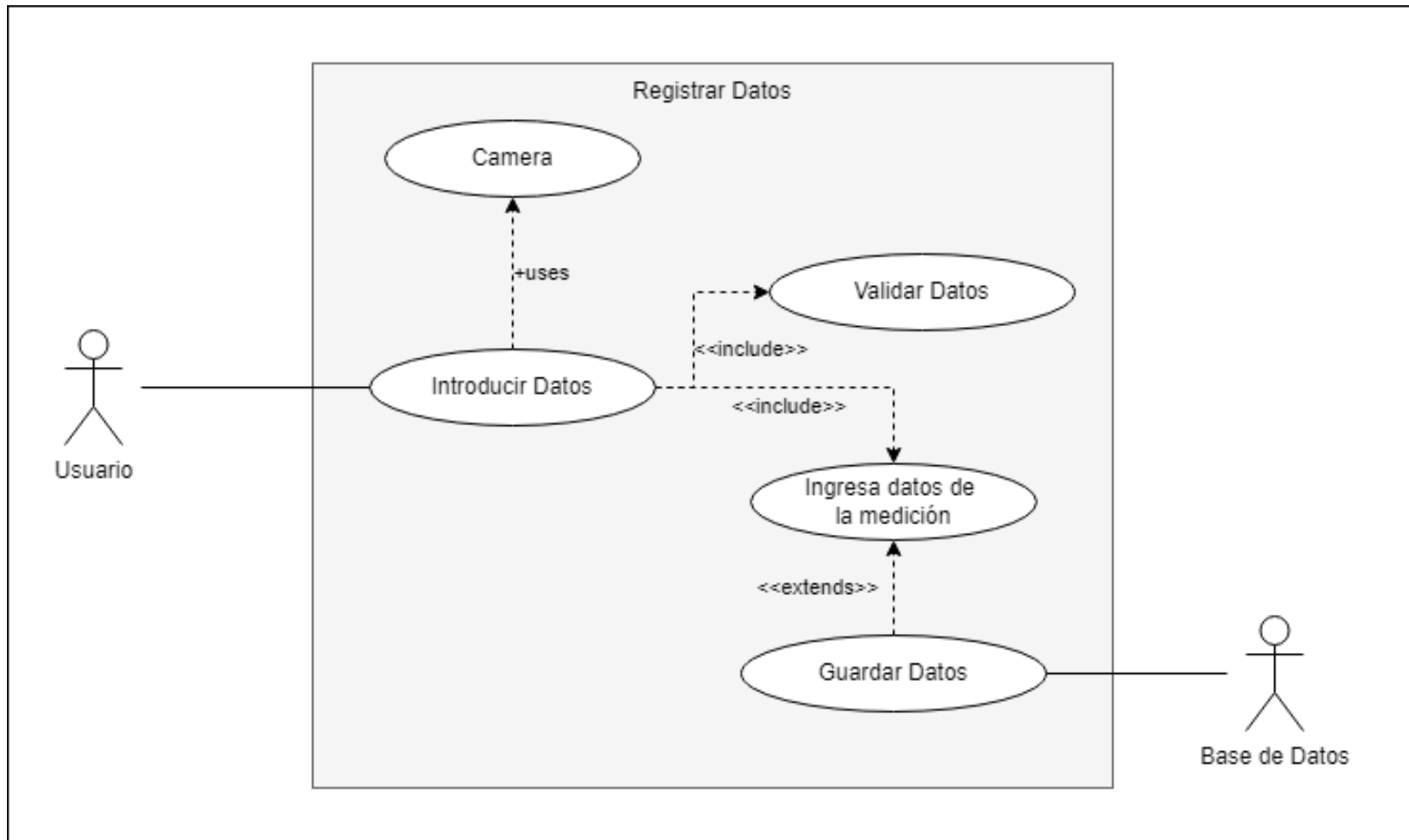


Figura 23: Diagrama caso de uso Registrar mediciones, Elaboración Propia

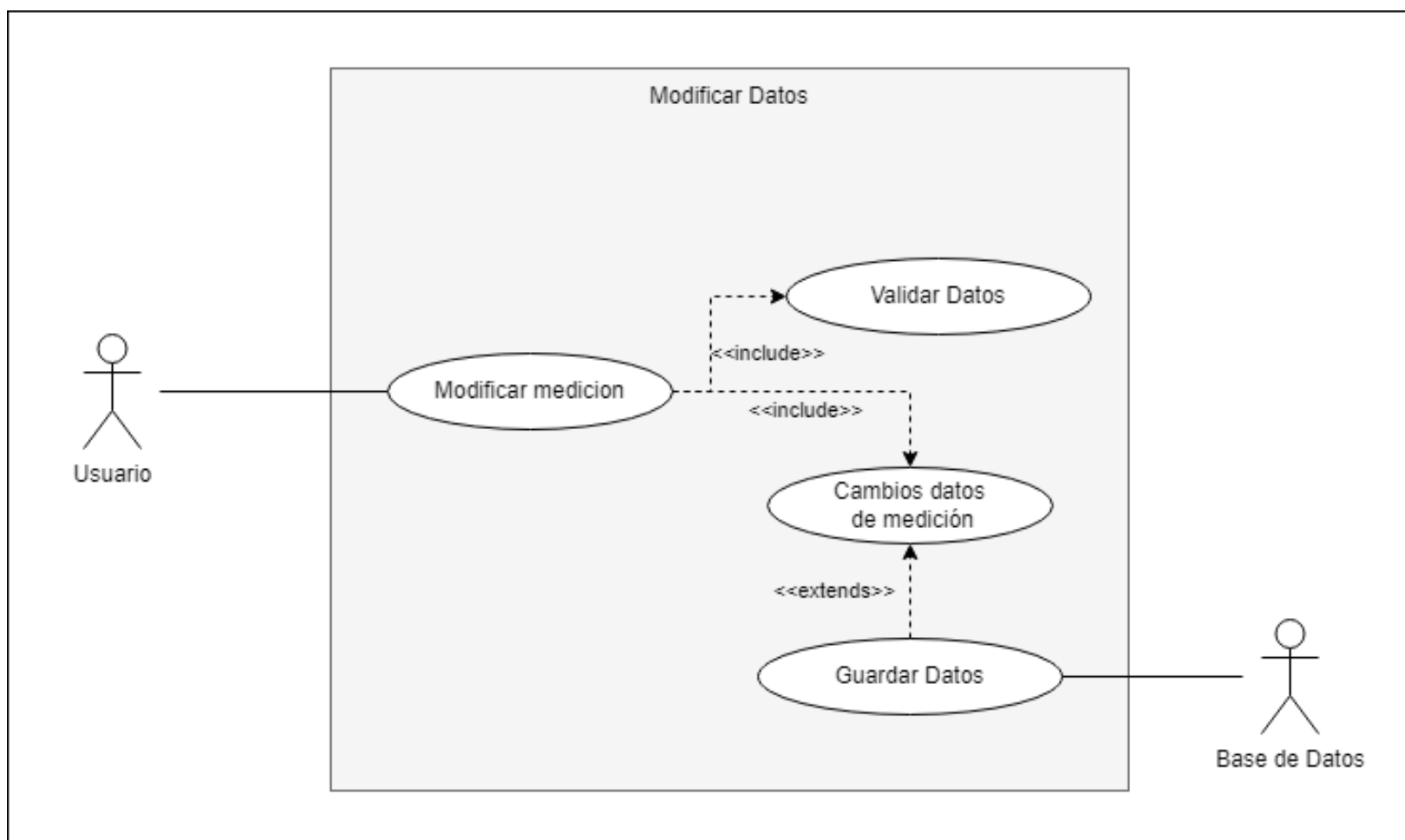


Figura 22: Diagrama caso de uso Modificar mediciones, Elaboración Propia

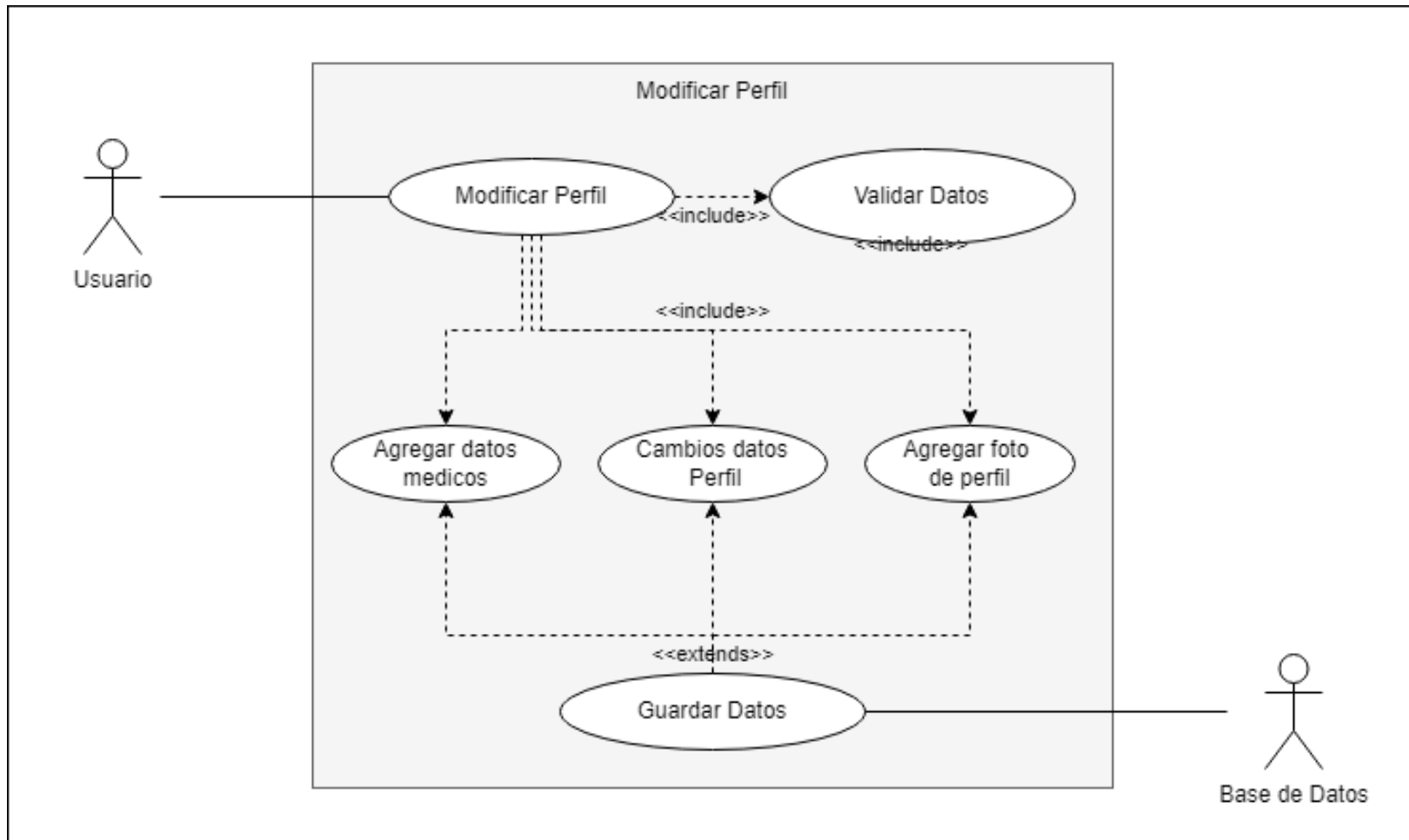


Figura 24: Diagrama caso de uso Modificar usuarios, Elaboración Propia

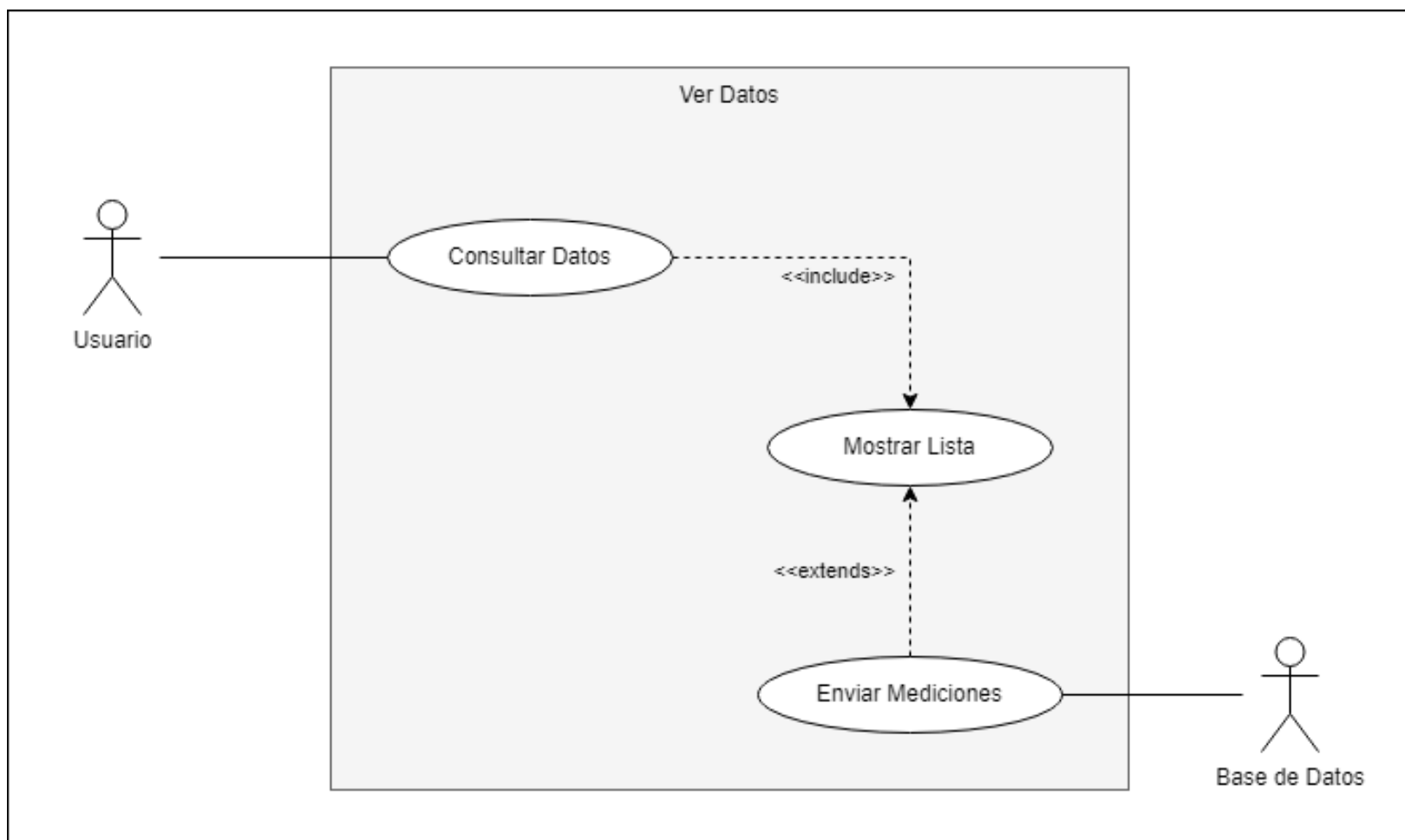


Figura 25: Diagrama caso de uso Ver datos, Elaboración Propia

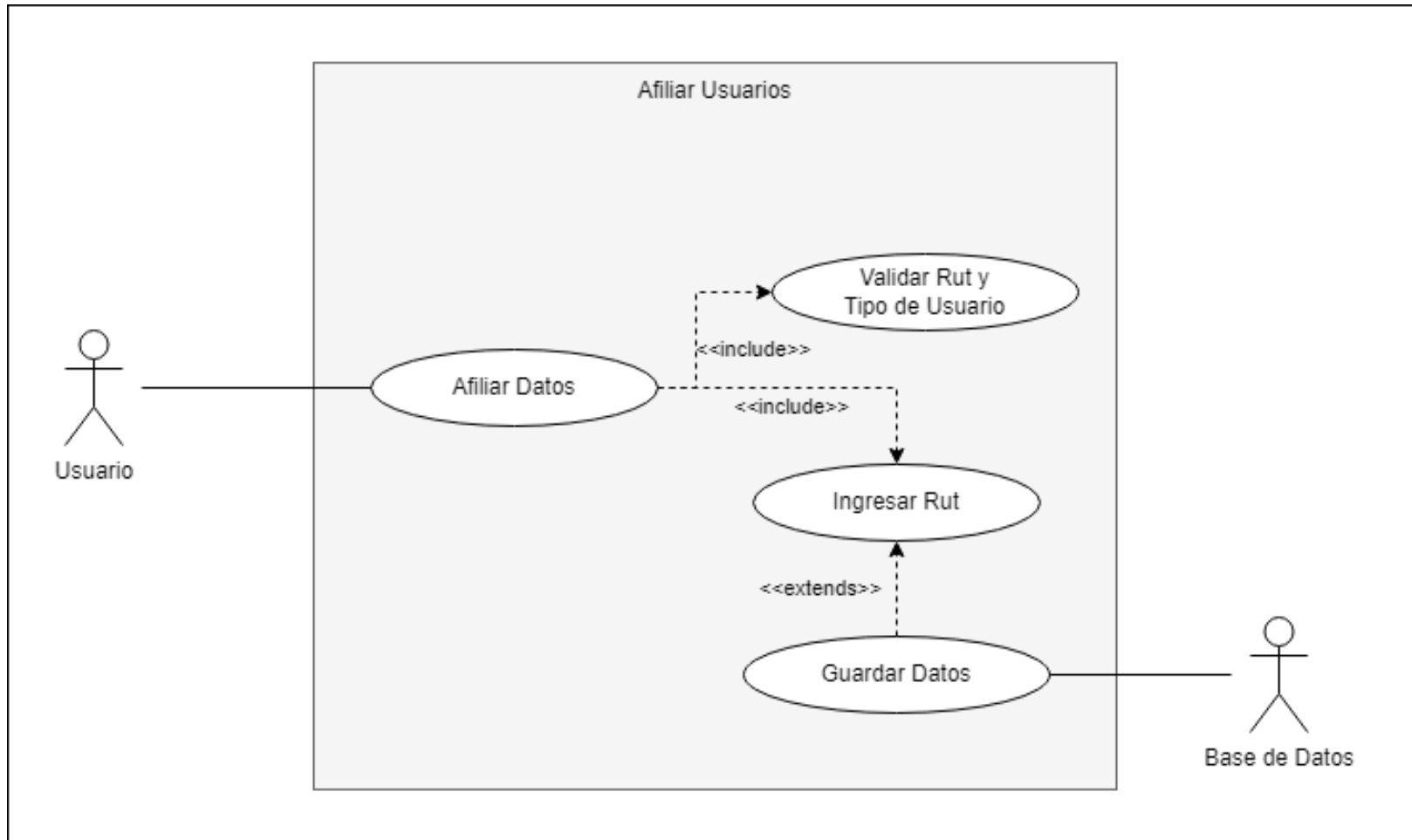


Figura 26: Diagrama caso de uso Afiliar usuarios, Elaboración Propia

4.4 Diagramas de Secuencia

Los siguientes diagramas de secuencia mostrados a continuación buscan modelar el funcionamiento del sistema, describiendo la interacción del usuario con el sistema mediante una línea de tiempo.

- Diagrama de Secuencia Registrarse: La Figura 27 muestra el diagrama correspondiente al registro de usuario, comenzando cuando el usuario accede a la aplicación y selecciona la opción de Registrarse, esto muestra el formulario que el usuario debe rellenar ingresando sus datos y el establecimiento del tipo de usuario, en caso de tener campos vacíos o la contraseña y su confirmación no son la misma, el usuario recibirá un mensaje a la interfaz, indicando el error correspondiente. Una vez completada esta etapa del proceso, la aplicación se comunica con las Bases de Datos, mandando la información del nuevo usuario a ingresar a ambas, recibida la confirmación de datos guardados, el usuario recibirá un mensaje de confirmación y la aplicación redirigirá al usuario a la interfaz de *login* con los datos de usuario que ingresó, permitiéndole ingresar a su *dashboard* presionando en el botón de *login*.

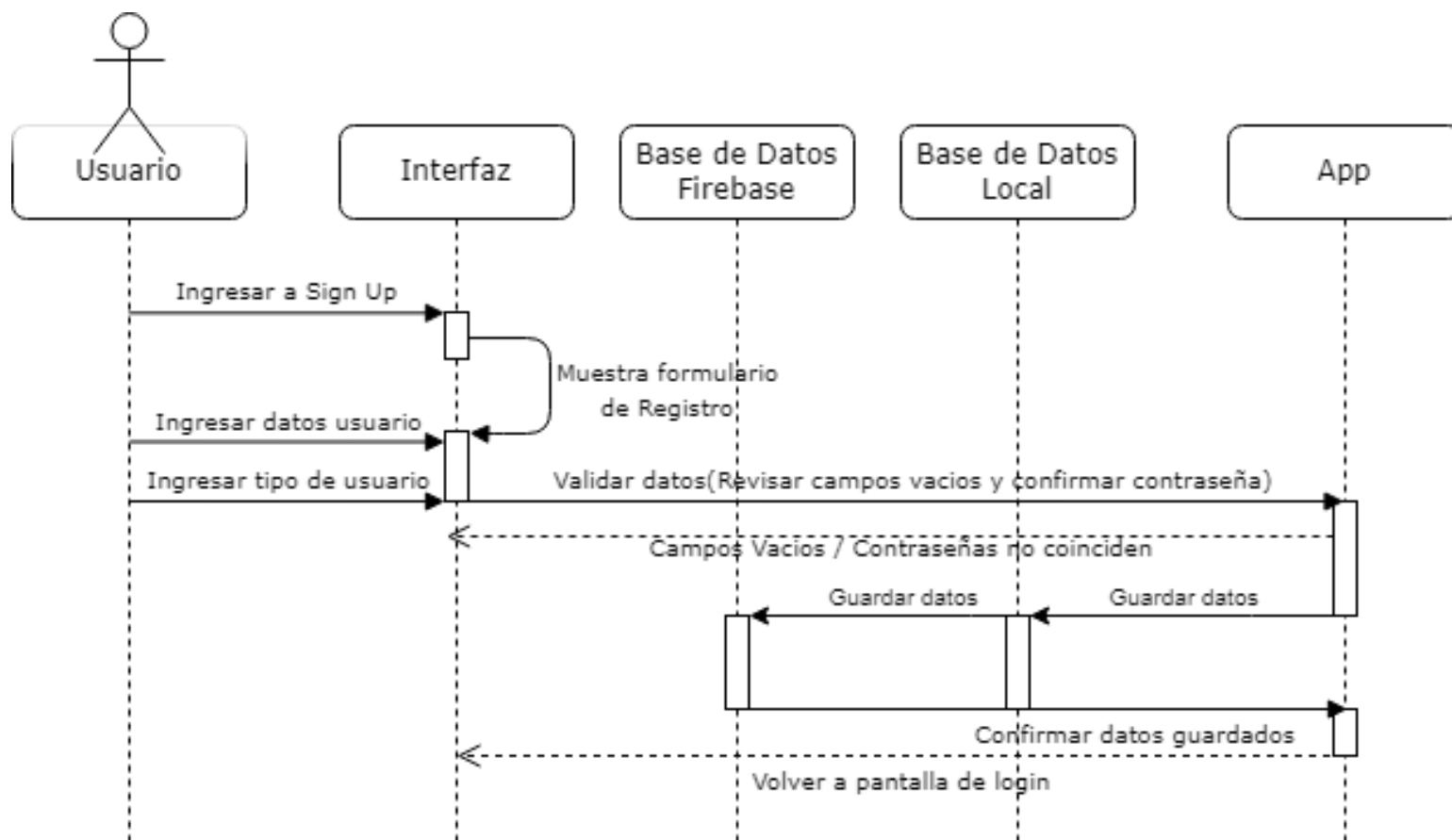


Figura 27: Diagrama de secuencia Registro, Elaboración Propia

- Diagrama de Secuencia Registro de Datos: La Figura 28 corresponde al proceso de introducir datos, este proceso inicia cuando se selecciona la opción de Agregar Datos, ya sea mediante datos o fotos, en cualquier caso, el usuario ingresará la información correspondiente, una vez se confirma que es correcta, se envía a la aplicación para realizar la validación de los datos ingresados y enviarla a ambas bases de datos para su almacenado, recibida la confirmación se le muestra al usuario sus datos guardados y se le redirige al *dashboard*.

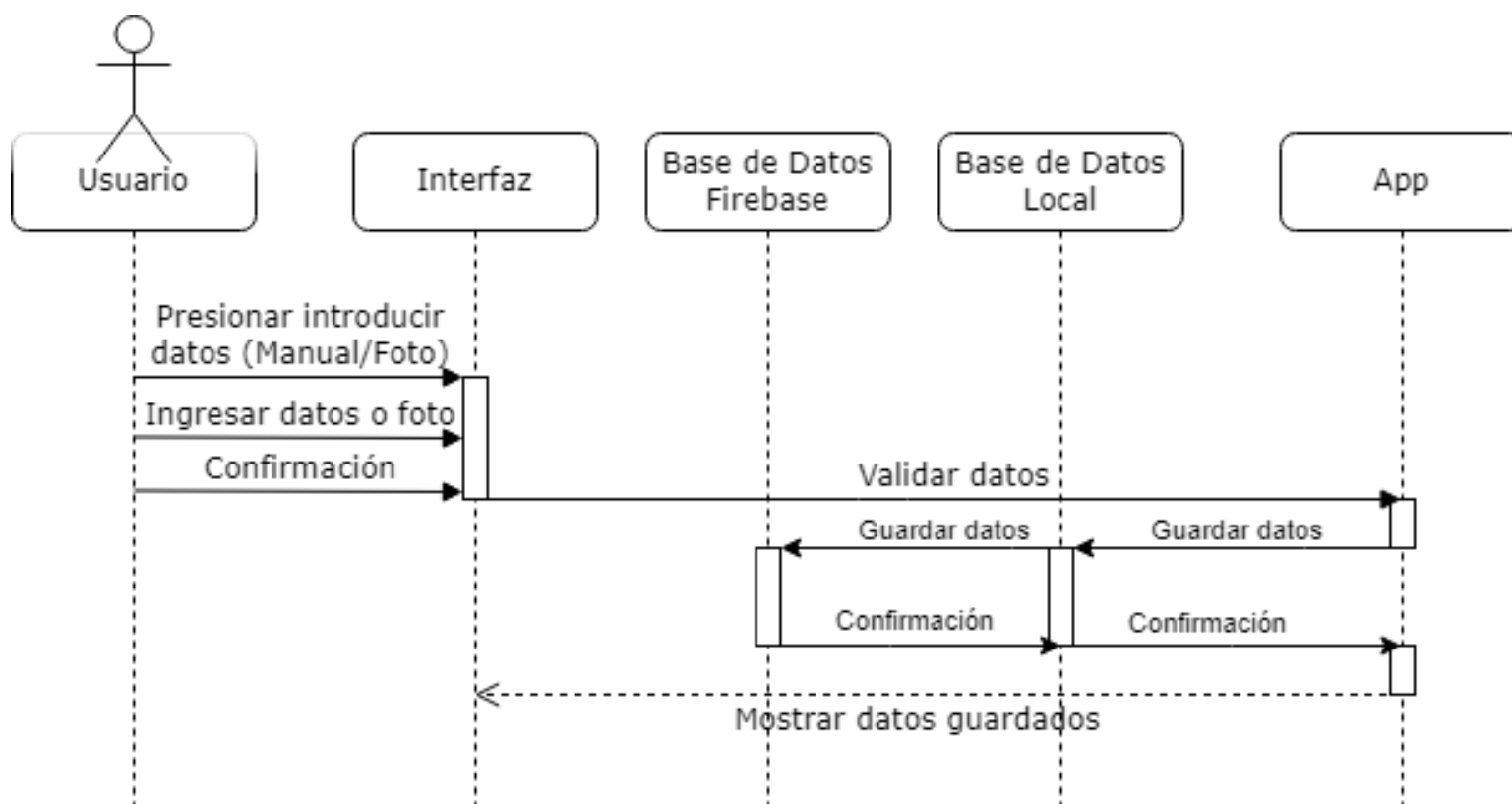


Figura 28: Diagrama de secuencia Registro mediciones, Elaboración Propia

- **Diagrama de Secuencia *Login*:** La Figura 29 muestra el diagrama correspondiente al Login de usuario, comenzando cuando se selecciona la opción de *Login*, el usuario debe ingresar sus credenciales, estas son enviadas a la aplicación, la cual consulta por el usuario en la base de datos local, de ser encontrado, las credenciales son válidas y por lo tanto, la aplicación se redirigirá al dashboard correspondiente. En caso de que la base de datos no encuentre al usuario se le enviará un mensaje por medio de la interfaz, indicando que su nombre de usuario o contraseña son inválidos.

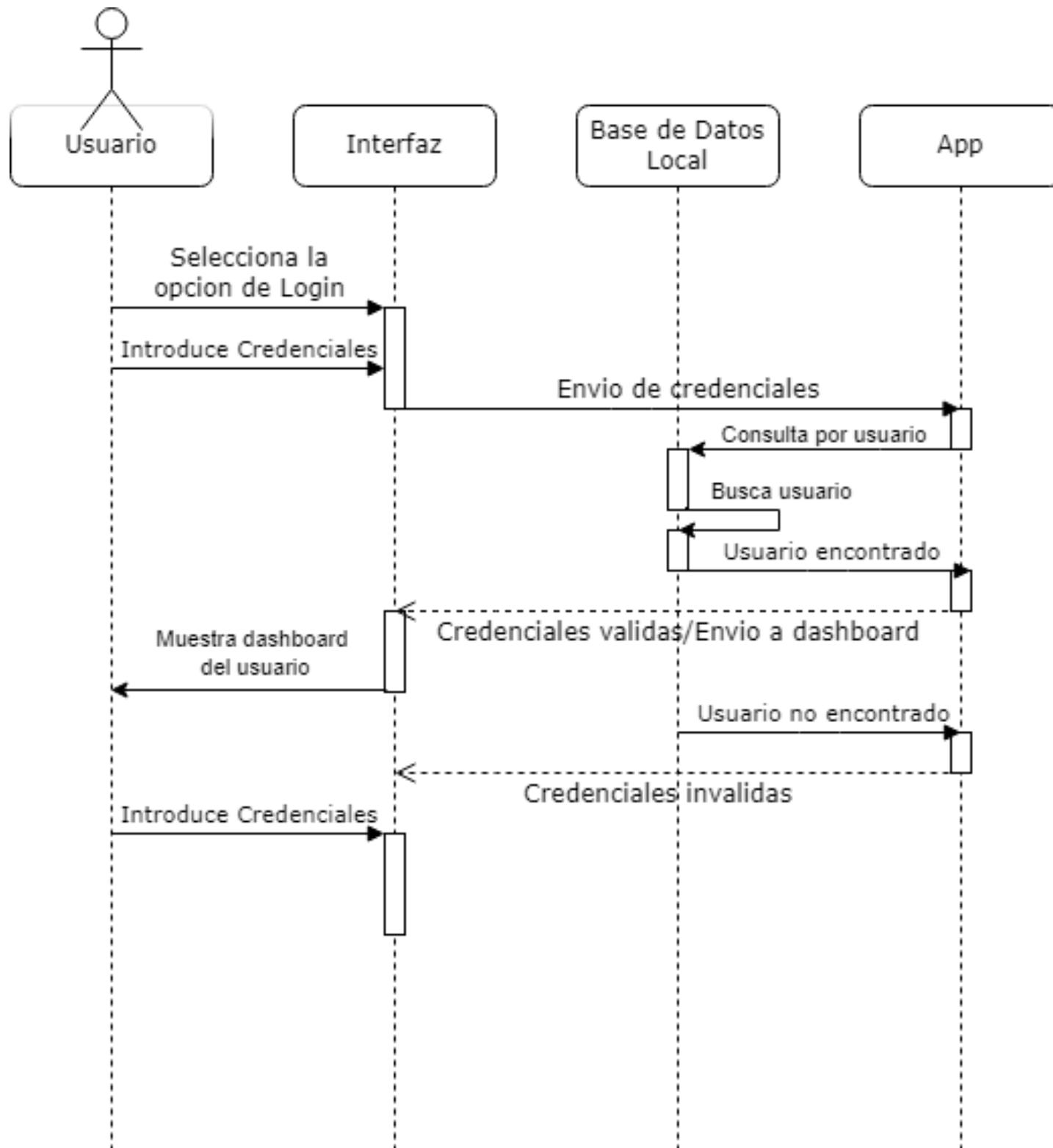


Figura 29: Diagrama de secuencia Login, Elaboración Propia

- Diagrama de Secuencia Conexión *Bluetooth*:** La Figura 30 corresponde al proceso de conexión con un dispositivo compatible mediante *Bluetooth*, comenzado con la selección de esta opción, se debe realizar una búsqueda mediante el bluetooth. El dispositivo transmitirá información a la aplicación y una vez se valide, esta sabrá de la presencia del dispositivo por lo que lo añadirá a la lista de dispositivos encontrado. El usuario seleccionará este para que la aplicación se sincronice enviando una solicitud al dispositivo, este realizará sus medidas correspondientes y transmitirá los valores a la aplicación, la cual mostrará los datos al usuario para que este pueda posteriormente guárdalos utilizando la misma secuencia de registro de datos ya explicada.

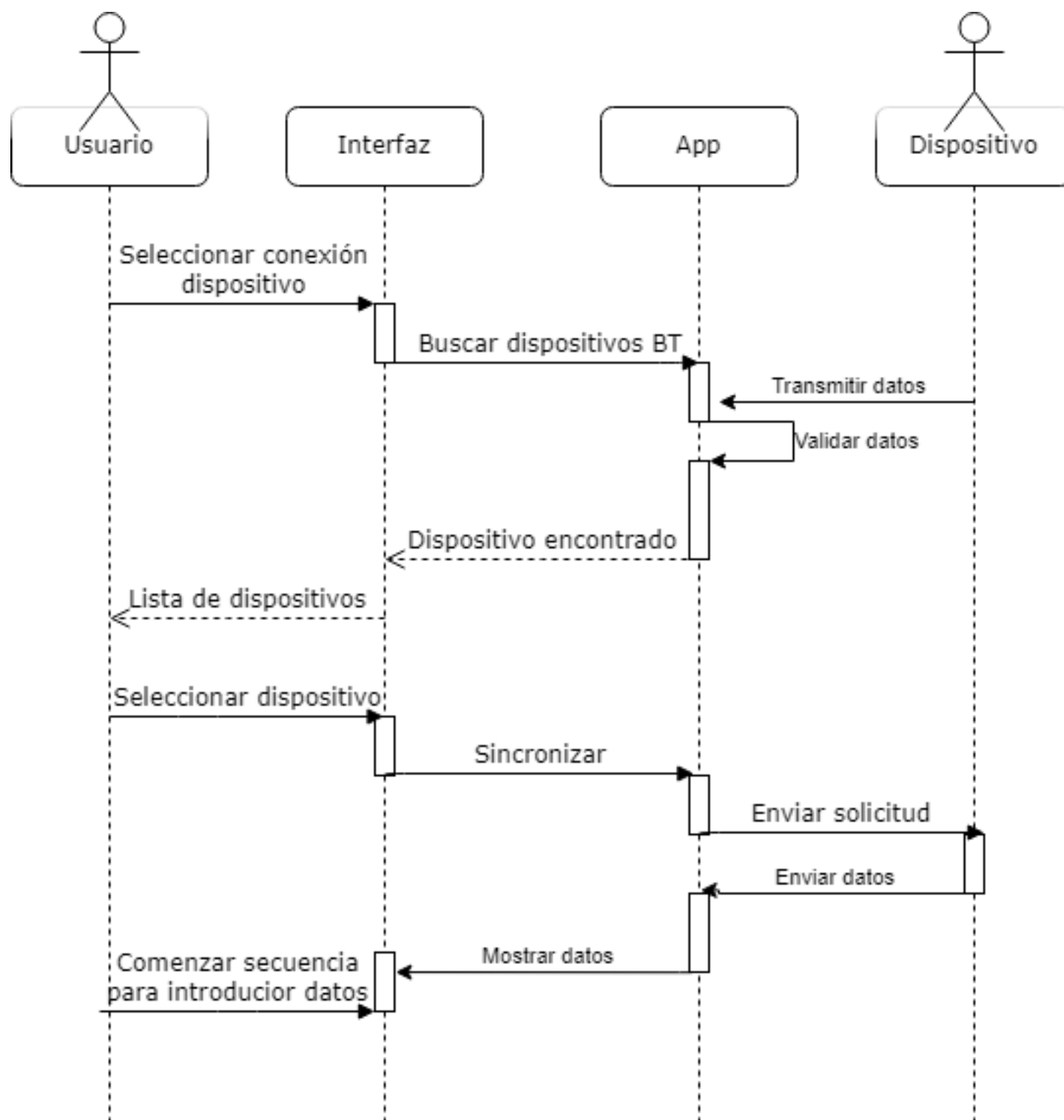


Figura 30: Diagrama de secuencia Conexión BT, Elaboración Propia

- Diagrama de Secuencia Modificar o agregar datos del usuario: La Figura 31 muestra el proceso de modificación o agregado de datos al perfil del usuario, al momento que se selecciona esta opción, la aplicación realizará una solicitud de los datos del usuario a la base de datos local, esta le enviará los datos a la aplicación para que puedan ser mostrados, se seleccionarán los campos a modificar o añadir. Una vez se confirmen todos los cambios realizados, mediante el ingreso de la contraseña, la aplicación validará los datos y se guardarán en ambas bases de datos, recibida la confirmación se redirigirá al usuario al dashboard.

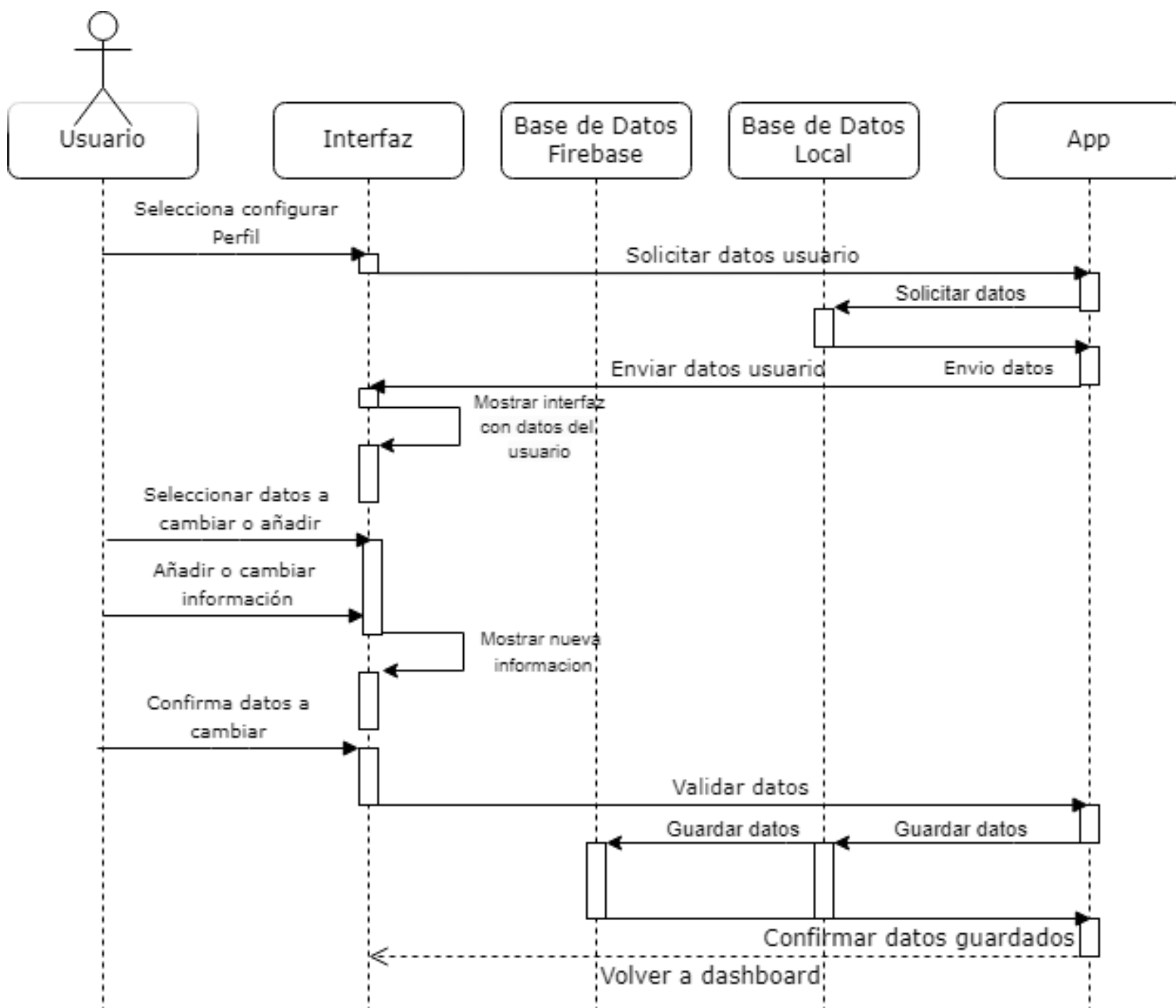


Figura 31: Diagrama de secuencia Modificar Usuario, Elaboración Propia

- Diagrama de Secuencia Ver registro de datos: La Figura 33 y Figura 32 corresponden al proceso de ver datos, este proceso inicia en el caso de los pacientes al seleccionar Todos los Registros, mandando una solicitud a la aplicación para obtener las mediciones del usuario, esta se le pide a la base de datos, obtenida esta información en le muestra al usuario a través de la interfaz. Para los otros tipos de usuario simplemente se debe escoger el paciente afiliado al cual se quiere ver la información antes de seleccionar la opción de Todos los Registros.

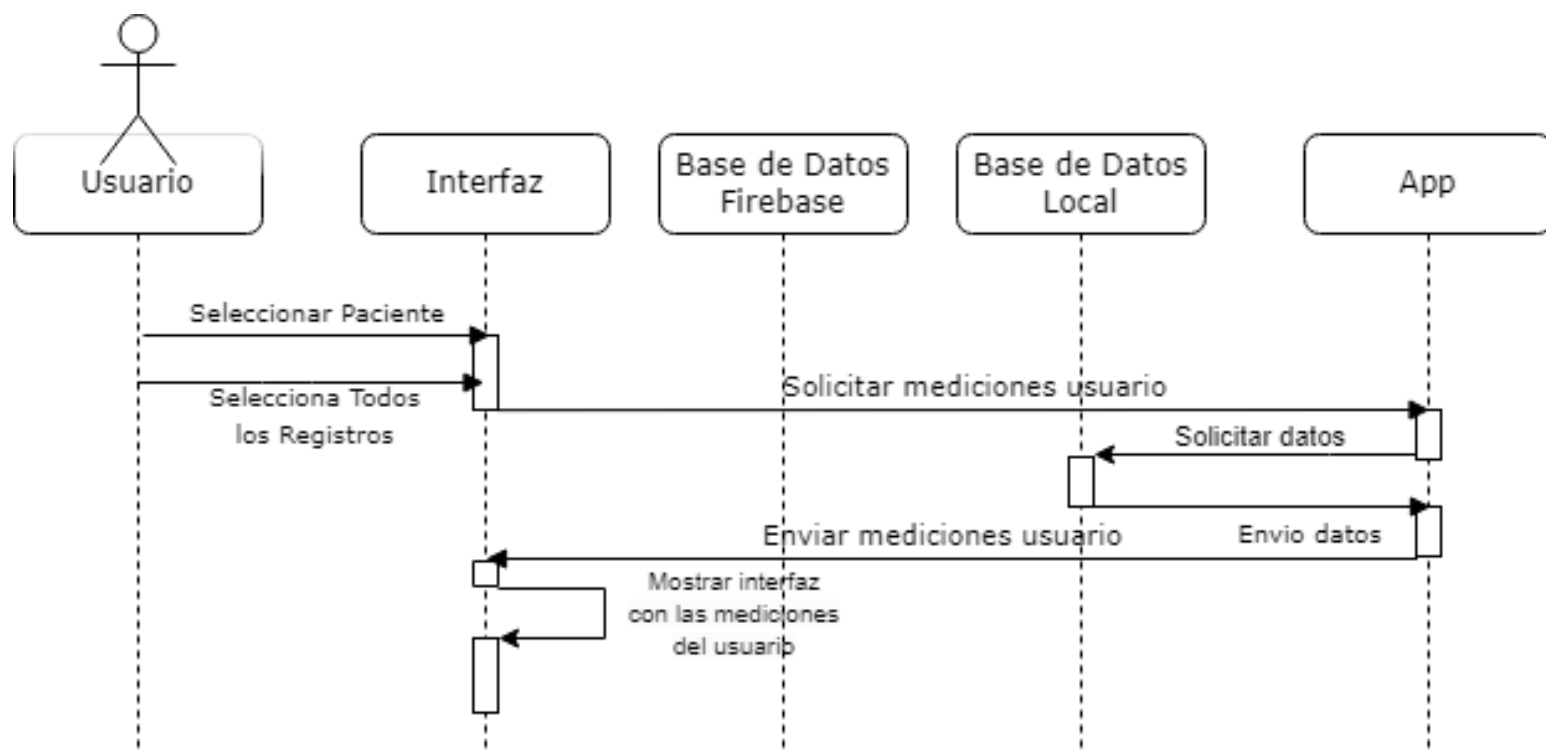


Figura 32: Diagrama de secuencia Ver datos, Elaboración Propia

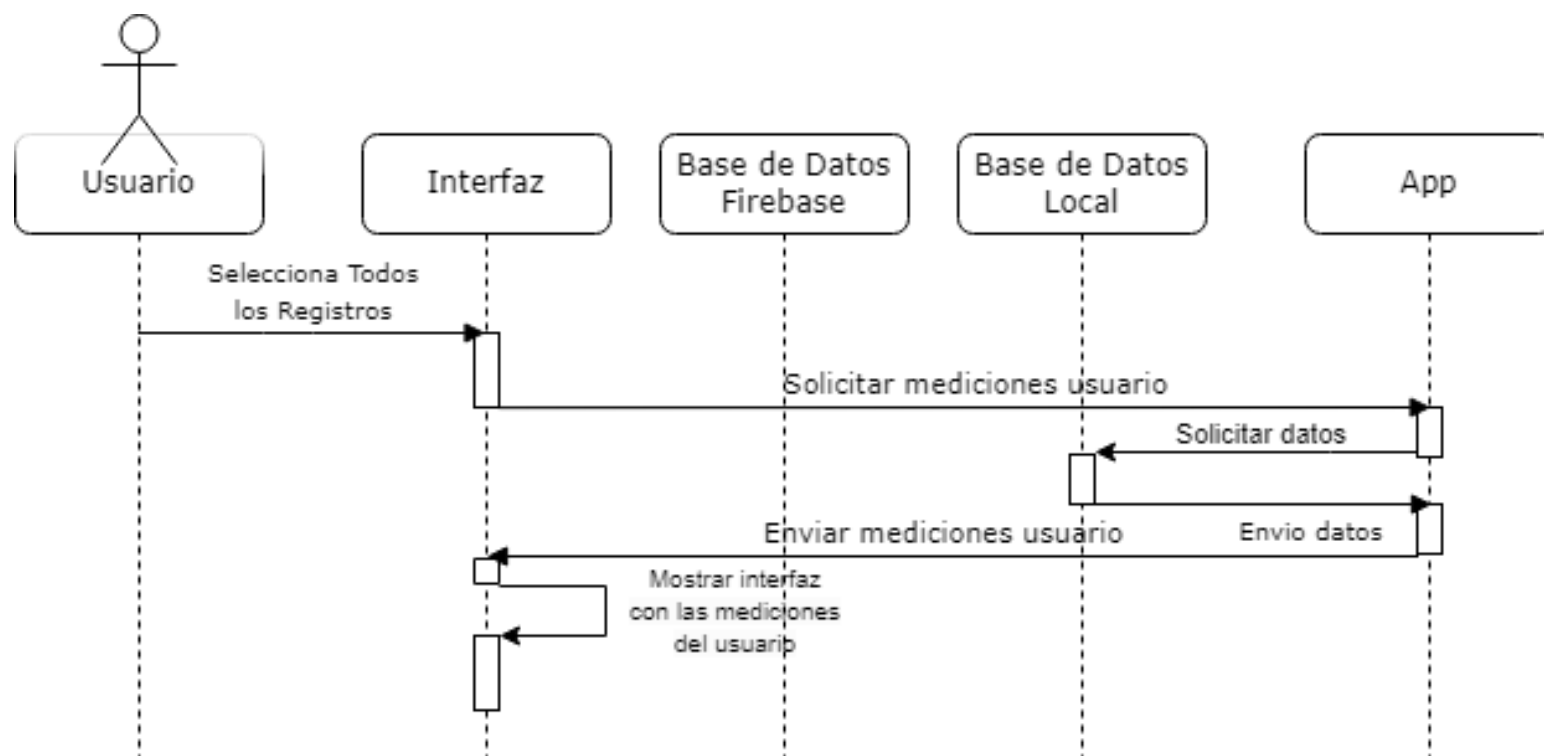


Figura 33: Diagrama de secuencia Ver datos, Elaboración Propia

- Diagrama de Secuencia Modificar o eliminar mediciones del usuario:** La Figura 34 muestra el proceso de modificación o eliminado de mediciones, este proceso inicia desde el final del proceso anterior, es decir, en la lista de todas las mediciones, en esta interfaz se debe seleccionar la medición a modificar, lo que mostrará al usuario los datos correspondientes para que se puedan realizar los cambios que se necesiten, la interfaz reflejará esta nueva información. De esta forma el usuario puede confirmar los cambios, enviando una solicitud para validar los datos a la aplicación, esta guardará los cambios en ambas bases de datos, recibida la confirmación el usuario volverá a la interfaz de todos los registros, si el usuario desea eliminar una de sus mediciones, debe seleccionarla, confirmar la eliminación, enviando la solicitud para ser retirada de las bases de datos. Finalmente, se mostrarán los cambios en la interfaz.

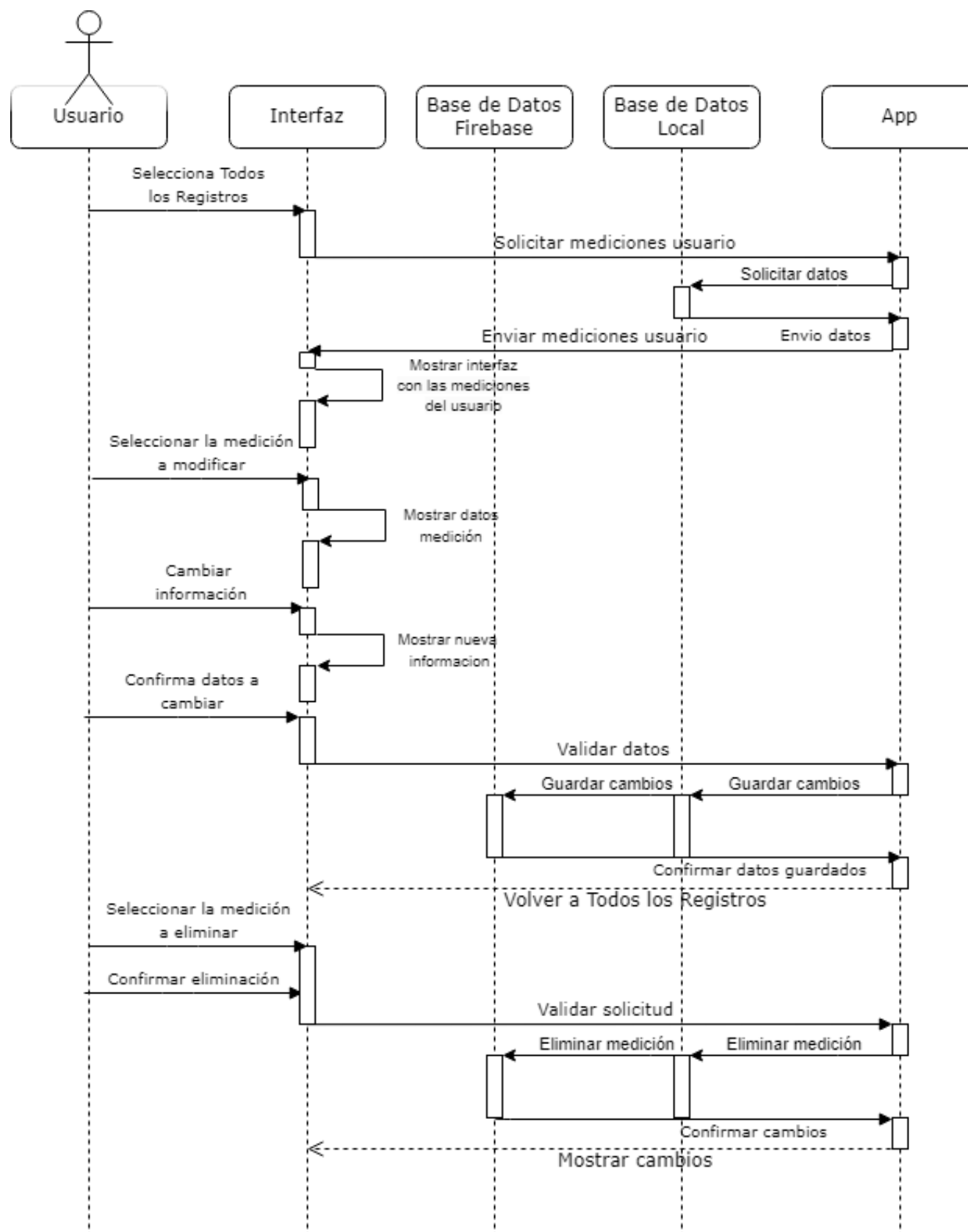


Figura 34: Diagrama de secuencia Modificar mediciones, Elaboración Propia

- Diagrama de Secuencia Afiliar usuarios:** La Figura 35 corresponde al proceso de afiliación de usuarios, este proceso inicia al seleccionar Afiliar Usuario, cambiando la interfaz para permitir al usuario ingresar un rut, la aplicación realizará una búsqueda del usuario en la base de datos con la información ingresada, si este usuario es encontrado, la aplicación determinará si es válido, en caso de serlo se mostrarán en la interfaz algunos datos del usuario encontrado como su nombre, email y tipo de usuario, para que se pueda realizar la confirmación y posteriormente se guarden los cambios en ambos perfiles, entregando un mensaje de confirmación al final, en caso de que el usuario no se encuentre o este no sea válido, se entregará un mensaje al usuario indicando que el rut o tipo de usuario es incorrecto.

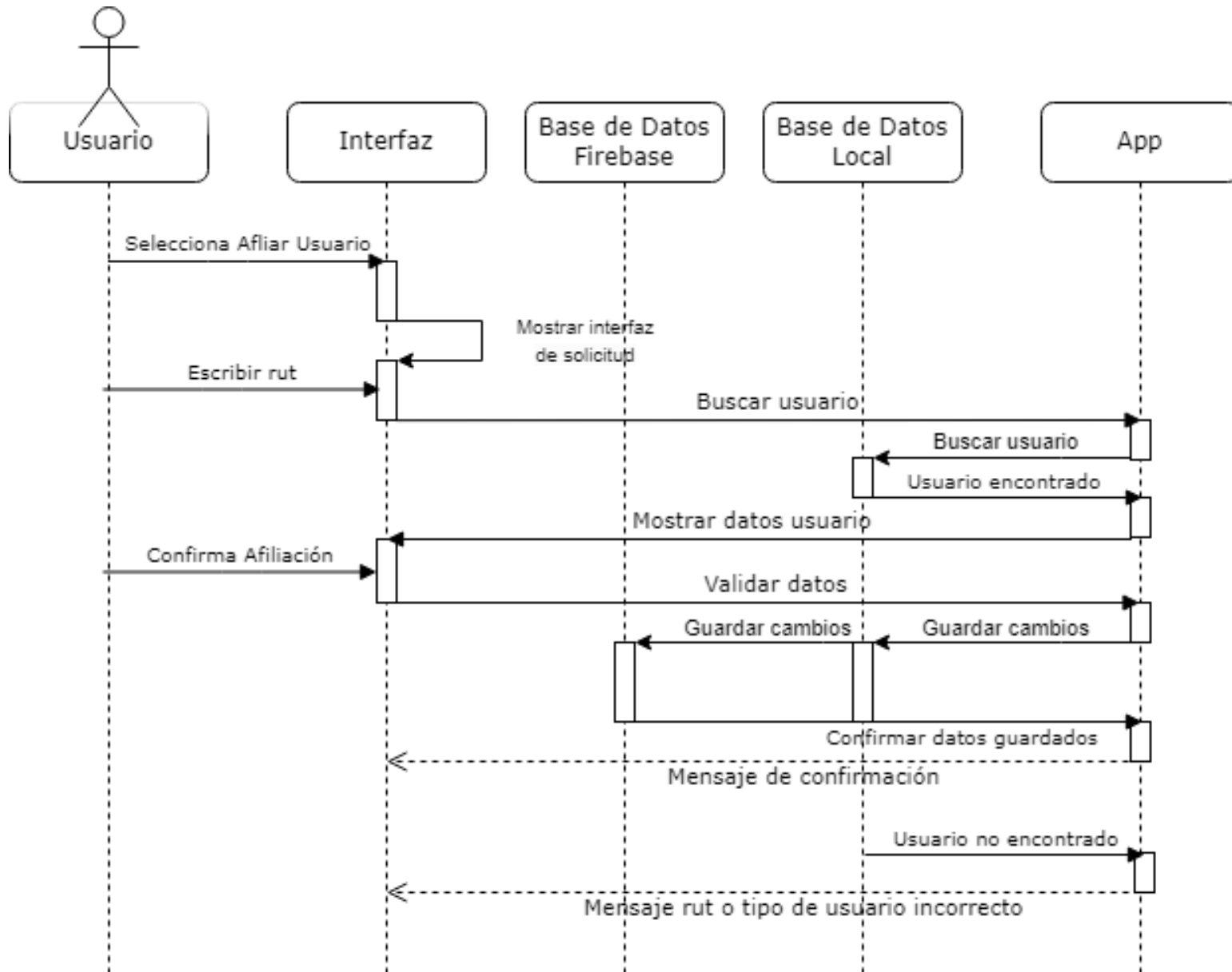


Figura 35: Diagrama de secuencia Afiliar usuario, Elaboración Propia

4.5 Diagrama de Clases

El siguiente diagrama muestra las clases del sistema, se pueden observar los distintos métodos que pertenecen a cada objeto y como se construyen, además de los atributos de cada clase.

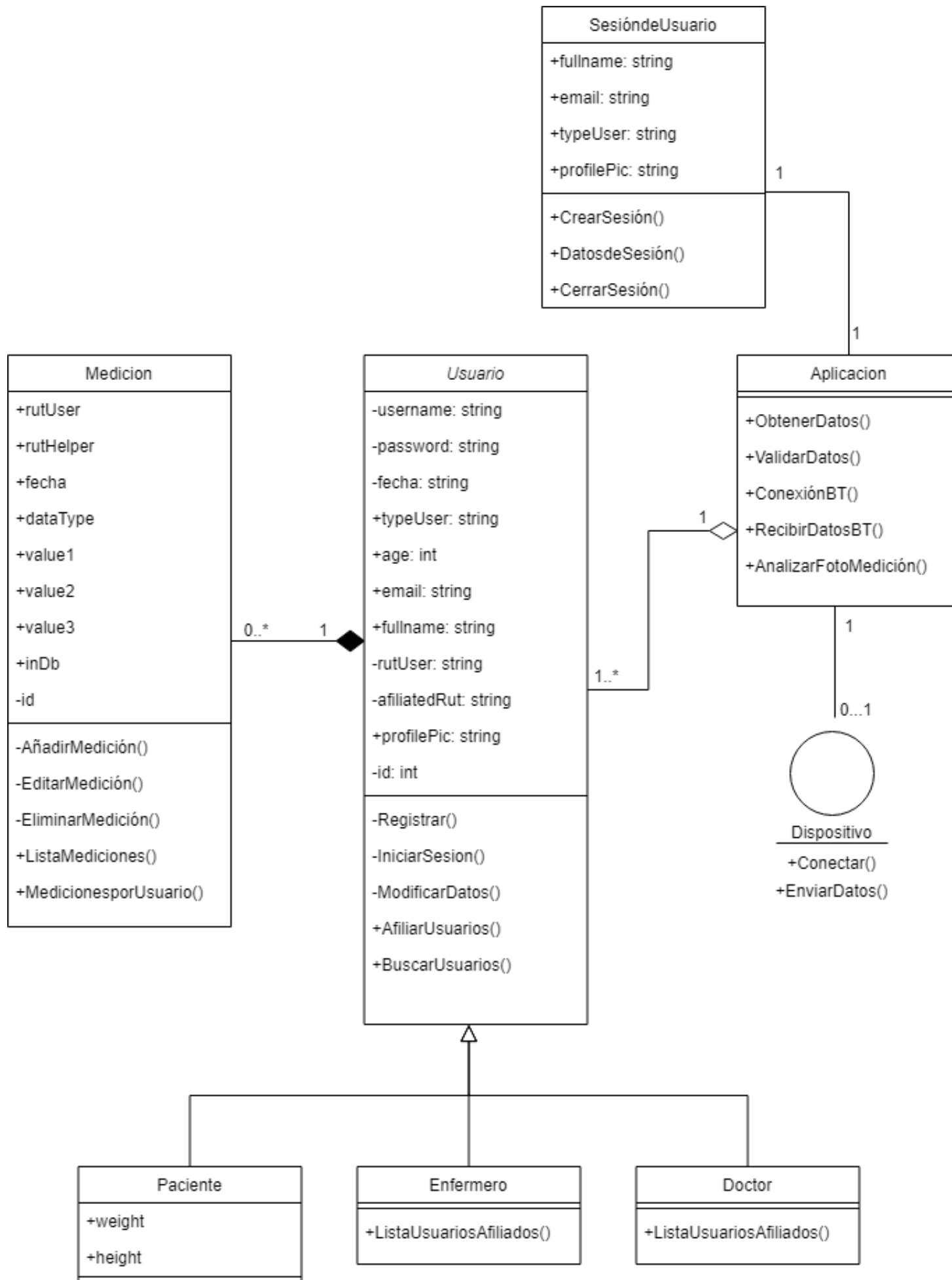


Figura 36: Diagrama de clases, Elaboración Propia

4.6 Diagrama de Actividades

La Figura 37 corresponde al diagrama de actividades general de la aplicación, la cual indica las interacciones de cada una de las acciones del usuario con la aplicación, desde que este abre la aplicación y comienza por registrarse, siempre y cuando no tenga una cuenta registrada previamente. Una vez iniciada la sesión, el usuario tendrá acceso a su dashboard desde donde podrá agregar mediciones, mediante fotos o de manera manual, además de ver sus últimos datos. A su vez tendrá acceso al menú lateral desde donde podrá ver todos sus registros, al igual que editarlos y eliminar los que quiera, ver, editar y agregar sus datos médicos personales, realizar la conexión bluetooth con el dispositivo con el cual va a interactuar la aplicación, afiliar usuario para compartir sus mediciones y cerrar sesión.

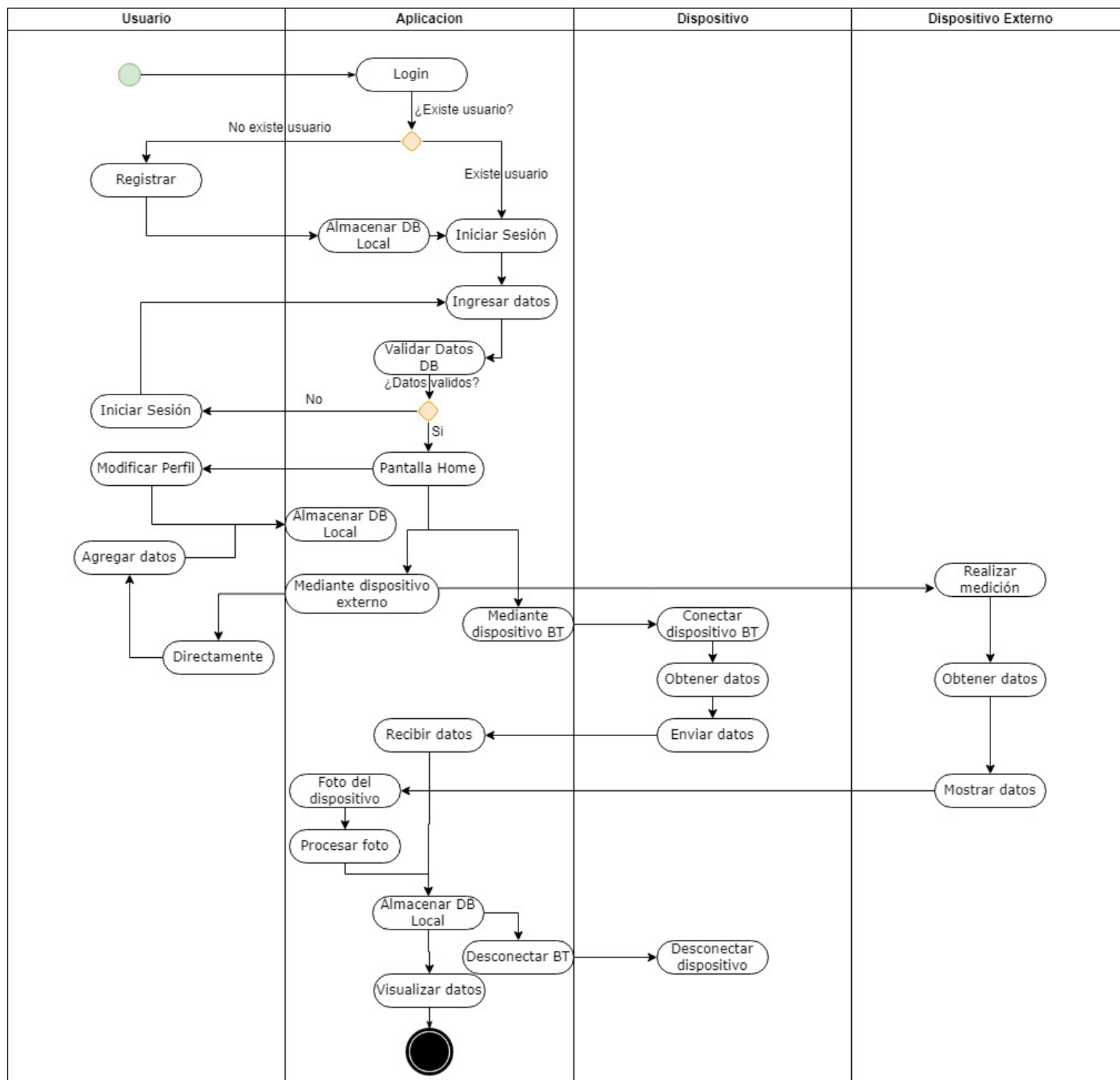


Figura 37: Diagrama de actividades, Elaboración Propia

4.7 Diagrama de Estados

La Figura 38 corresponde al diagrama de estados de la aplicación, este indica los cambios de estado por los cuales la aplicación pasará durante su uso, así como los procesos que debe realizar cuando se encuentra en los diferentes estados disponibles.

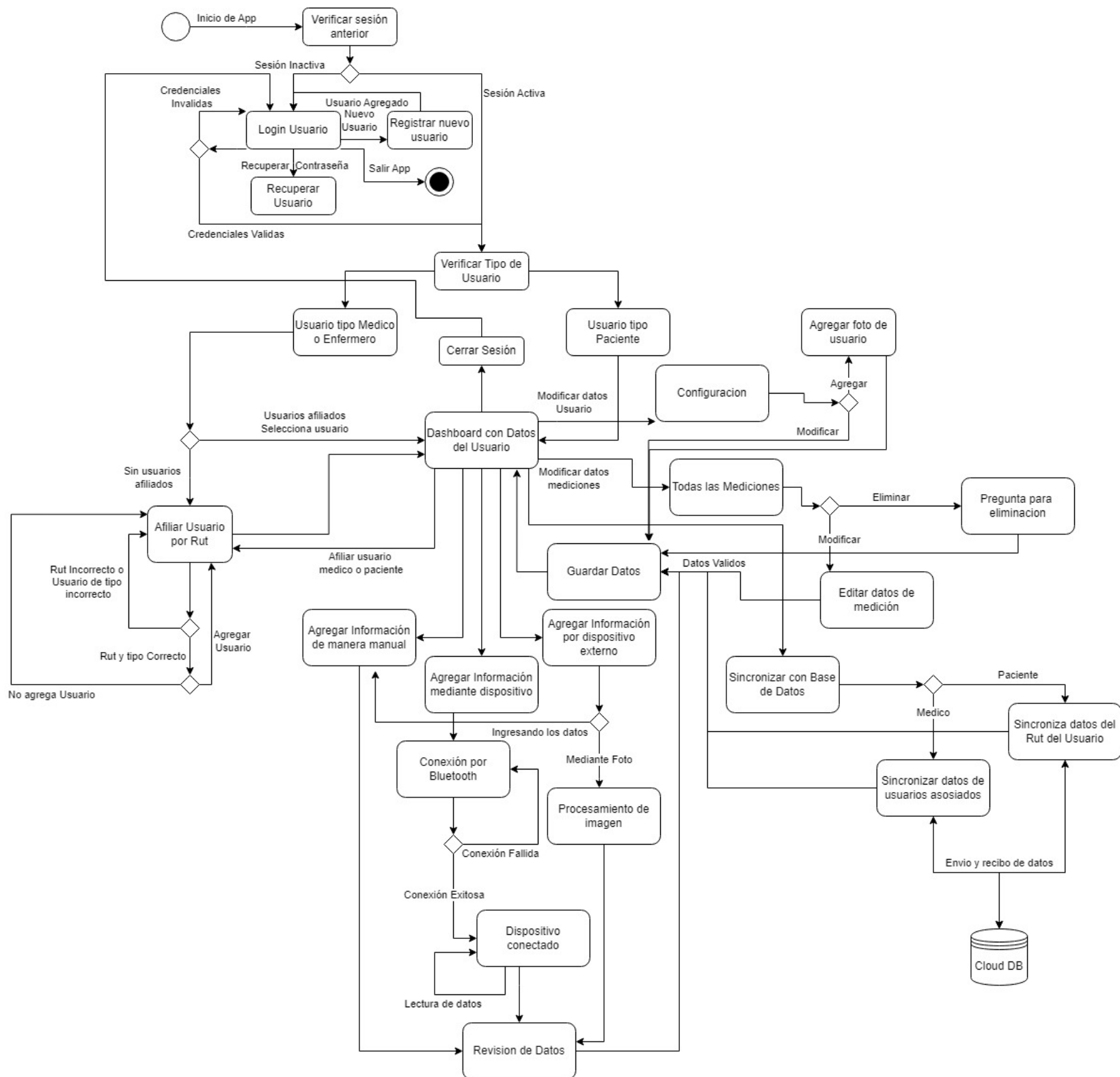


Figura 38: Diagrama de estados, Elaboración Propia

4.8 Diseño y descripción de interfaces

4.8.1 Interfaz de *Login* y *Sign Up*

Al iniciar la aplicación se mostrará la Figura 39, que contiene un texto de bienvenida al usuario y la opción de dirigirse ya sea al *Login* para que el usuario pueda ingresar con las credenciales de su cuenta registrada, Registrarte para que pueda rellenar sus datos en el formulario y seleccionar el tipo de usuario, creando así un usuario en la base de datos, además está la opción de sincronizar para que el usuario pueda actualizar su información con la base de datos.



Figura 39: Interfaz de Bienvenida, Elaboración Propia



Figura 41: Interfaz de Login, Elaboración Propia



Figura 40: Interfaz de Registro, Elaboración Propia

4.8.2 Interfaz *Dashboard*

Una vez en el *Dashboard* se tiene acceso a todas las diferentes funcionalidades de la aplicación, como añadir datos en su forma manual o mediante una foto, además de poder ver las últimas 4 mediciones realizadas por el usuario, en caso de ser del tipo de usuario Doctor o Asistente, se tiene acceso a la selección de paciente, que le permitirá ver la información asociada a sus usuarios afiliados.

Desde aquí también se tiene acceso al menú lateral, donde se puede seleccionar las siguientes opciones:

- Ver Registro (Figura 44): Que contiene todas las mediciones del usuario, además de dar acceso a las opciones de editar o eliminar la medición seleccionada.
- Afiliados (Figura 45): Esta opción permite al usuario buscar algún perfil por medio de Rut, si se encuentra un usuario del tipo correcto, es decir, en caso de ser paciente, este usuario debe ser del tipo doctor o asistente; en caso de encontrar a un usuario compatible, se mostrará el nombre, tipo de usuario y mail, además de su foto de perfil en caso de que este tenga una, justo con la aparición del botón para confirmar el afiliado a ese usuario.
- Configuración (Figura 43): Esta opción permite al usuario agregar y editar su información personal, para guardar los cambios se debe ingresar su contraseña.
- Conectar BT (Figura 42): Contiene la opción de conectarse a un dispositivo compatible por medio de bluetooth, permitiéndole al usuario realizar mediciones mediante el dispositivo.
- Cerrar Sesión: Se eliminan los datos de la sesión del usuario, además vuelve al menú principal.



Figura 47: Interfaz de Dashboard, Elaboración Propia



Figura 46: Interfaz de Dashboard Medico, Elaboración Propia



Figura 45: Interfaz de Afiliados, Elaboración Propia



Figura 44: Interfaz de Ver registro, Elaboración Propia



Figura 43: Interfaz de Configuración, Elaboración Propia



Figura 42: Interfaz de Conexión BT, Elaboración Propia

4.8.3 Interfaz Añadir Mediciones

Añadir Mediciones Manual (Figura 50): Esta interfaz corresponde a un formulario con los datos básicos para registrar una medición, la fecha y el rut del usuario se pondrán de manera automática.

Añadir Mediciones Fotografía: Corresponde a 3 interfaces comenzando con la Figura 48, donde al usuario se le pedirá seleccionar el tipo de dispositivo a ingresar, además si la foto de este vendrá de la galería o mediante la cámara del teléfono. La galería y la cámara utilizadas provienen de la interfaz del mismo sistema operativo, seleccionada la foto se pasa a la Figura 49 donde el usuario podrá mover y recortar la foto para que solo aparezca la información solicitada, finalmente el formulario de añadido de datos manual auto relleno con la información obtenida en la foto.



Figura 50: Interfaz de añadir manual, Elaboración Propia



Figura 49: Interfaz de *cropeo* de foto, Elaboración Propia



Figura 48: Interfaz de añadir foto, Elaboración Propia

4.9 Base de Datos Firebase

Para la creación de esta base de datos se debe ingresar mediante la consola de Firebase e iniciar un proyecto, ingresado al proyecto se encuentra la opción de *realtime database*, donde se nos dará la opción de iniciar la creación, terminado el proceso se debe conectar la aplicación a Firebase, para esto, en las opciones de vinculación se selecciona Android, se ingresará el nombre del paquete de Android, esto permitirá descargar un archivo *google-services.json*, se agregará en la carpeta del proyecto, finalmente se deben agregar las dependencias a la aplicación.

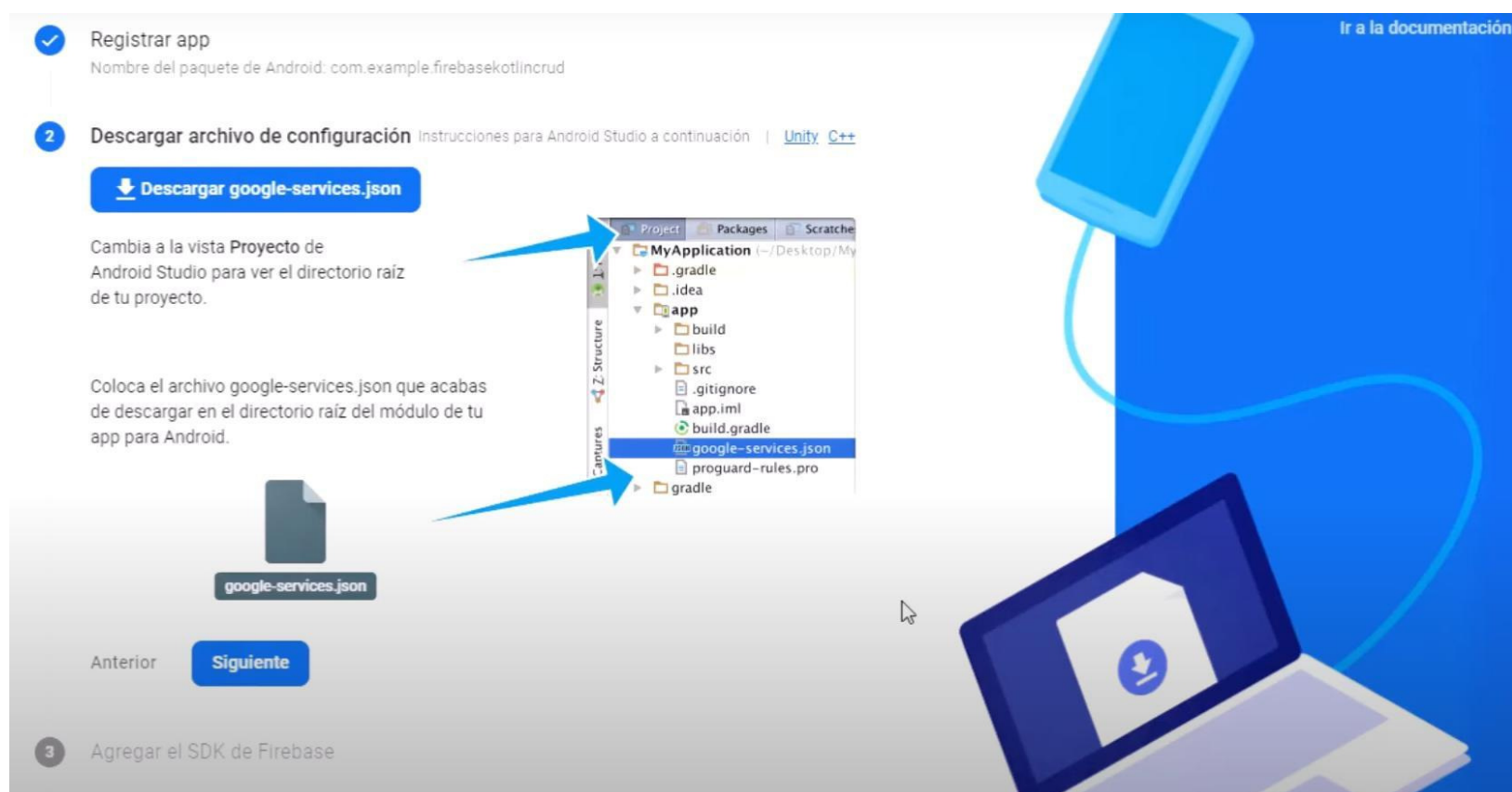


Figura 51: Conexión Android – Firebase , <https://console.firebase.google.com>

Terminando la configuración y con la base de datos creada se pueden acceder a las diferentes funciones, principalmente se utilizará esta base de datos como un respaldo de la local, además de almacenar los datos que agregan de diferentes dispositivos, de manera que si el usuario nota la ausencia de algún dato en la pantalla principal tiene la opción de sincronizarse con la base de datos de Firebase, esto se hace además de manera automática, la primera vez que el usuario usa la aplicación. De esta forma se tiene la misma base de datos local en todos los diferentes usuarios y dispositivos.

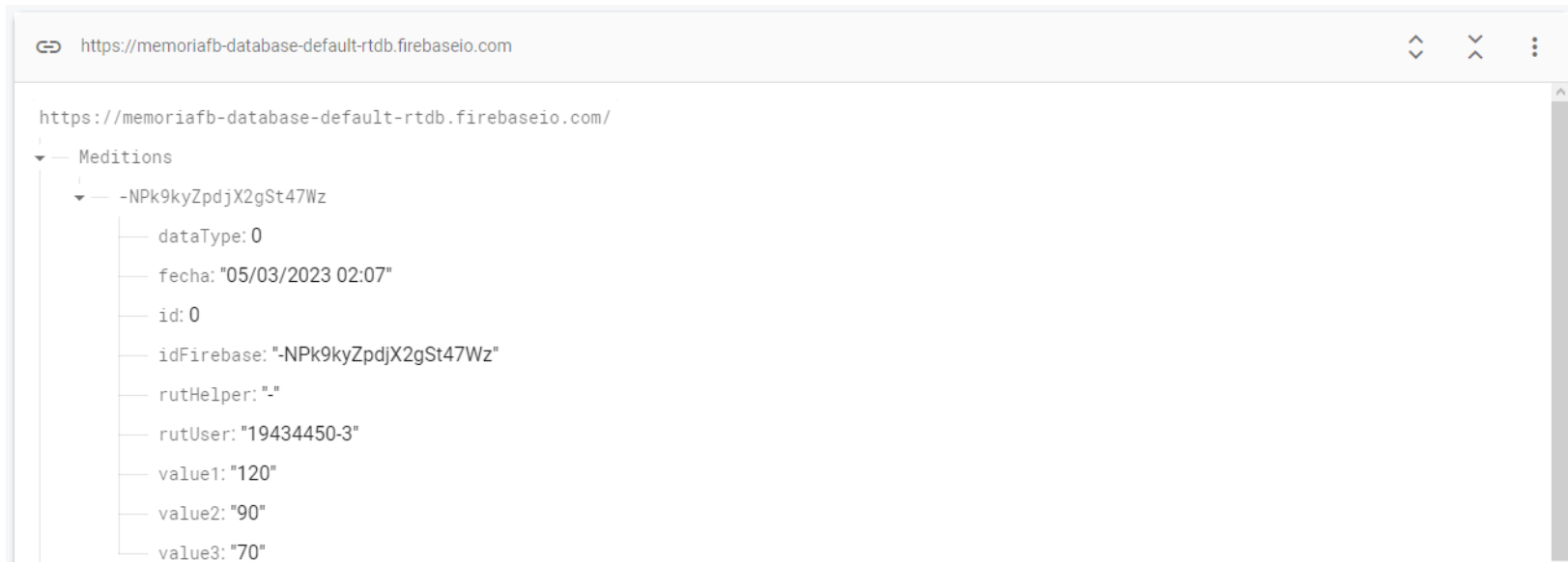


Figura 52: Base de datos Firebase, <https://console.firebase.google.com>

El CRUD de Firebase se realiza justo después de cada una de las entradas en la base de datos local, mediante una referencia a la base de datos, donde se obtendrá la instancia y se ingresa el nombre de la referencia, en caso de ingresar datos se crea una id de Firebase, que se utiliza para realizar una modificación o eliminación del dato.

Para sincronizar la base de datos durante la primera ejecución se revisa si hay algún perfil creado, en caso de no encontrar ninguno se comienza a sincronizar con Firebase, el usuario además tiene la opción de seleccionar sincronizar, se revisarán las id Firebase que estén ingresadas en la base de datos local y se agregarán las faltantes.

```
userViewModel.users.observe( owner: this) { it: List<User>!
    if (it.isEmpty()) {
        userList = listOf()
    } else {
        userList = it
    }
    if (!userList.isEmpty()) {
    } else {
        Toast.makeText(
            applicationContext,
            text: "El primer uso de la aplicacion requiere de sincronizacion con la base de datos",
            Toast.LENGTH_SHORT
        ).show()
        //make user wait
        dbRefUser.addValueEventListener(object : ValueEventListener {
            override fun onDataChange(snapshot: DataSnapshot) {
                if (snapshot.exists()) {
                    for (users in snapshot.children) {
                        val username = users.child( path: "username").getValue(String::class.java)!!
                        val password = users.child( path: "password").getValue(String::class.java)!!
                        val fecha = users.child( path: "fecha").getValue(String::class.java)!!
```

Figura 53: Código sincronización inicial, Elaboración Propia

4.10 Implementación de *Login/Sign Up*

Para realizar el *login* del usuario se utiliza el modelo de Base de datos *Room* para la validación, por lo que con el nombre de usuario y contraseña se hace una solicitud al *viewModel* mediante una función, siendo este ultimo el intermediario entre la base de datos local y la aplicación, los datos recopilados del usuario estarán en la variable *user* que antes correspondía a un usuario vacío, por lo que si este deja de serlo el usuario podrá pasar a su dashboard

```
binding.buttonLogin.setOnClickListener{ it: View!
    val username = binding.textUsername.text.toString()
    val pass = binding.textPassword.text.toString()
    var user:User = User( username: "", password: "", fecha: "", typeUser: "", age: "", email: "", fullname: "", rutUser: "", afiliatedRut: "", profilePic: null)
    if(stringCheck(username)&&stringCheck(pass)){
        val result = lifecycleScope.launch(Dispatchers.IO) { this: CoroutineScope
            user = userViewModel.getUser(username, pass)
        }
    }
}
```

Figura 54: Código Login, Elaboración Propia

Para el *SignUp* se recopilan todas las variables del formulario para poder revisar si estas tienen algún valor asociado, revisado esto, se comparan la contraseña con su confirmación. Finalmente, y si está correcto, se pasa a la creación del usuario en la base de datos, se notifica al usuario que su cuenta se ha creado correctamente y se transporta a la pantalla de Login con sus datos registrados para que pueda ingresar a la aplicación.

```
binding.buttonSignUp.setOnClickListener{ it: View!
    val username:String = binding.textUsername.text.toString()
    val password:String = binding.textPassword.text.toString()
    val confPassword:String = binding.textConfirmPassword.text.toString()
    val age:String = binding.textAge.text.toString()
    val email:String = binding.textEmail.text.toString()
    val fullname:String = binding.textFullname.text.toString()
    val rutUser:String = binding.textRut.text.toString()
    var afiliatedRut:String = binding.textRutPaciente.text.toString() + ";"
    if(!stringCheck(afiliatedRut)){
        afiliatedRut=""
    }
    if(fullStringCheck(username,password, typeUser, age, email, fullname, rutUser)){
        if (password==confPassword) {
            val user = User(
                username,
                password,
                current,
                typeUser,
                age,
                email,
                fullname,
                rutUser,
                afiliatedRut,
                profilePic: null
            )
            userViewModel.addUser(user)
            val coor = Bundle()
            coor.putString("username",username)
            coor.putString("password",password)
            Toast.makeText(applicationContext, text: "Usuario creado correctamente", Toast.LENGTH_SHORT).show()
            val intent = Intent( packageContext: this, Login::class.java)
            intent.putExtras(coor)
            startActivity(intent)
        }else{
            //add error
            Toast.makeText(applicationContext, text: "Error Contraseñas distintas", Toast.LENGTH_SHORT).show()
        }
    }
}
```

Figura 55: Código SignUp, Elaboración Propia

4.11 Implementación de funciones

4.11.1 Afiliar Usuarios

Las funciones fundamentales de esta opción son la búsqueda que usuarios y posterior actualización de sus datos. La función de buscar usuarios utiliza el rut ingresado y busca las coincidencias en toda la base de datos, si se encuentra un usuario se realiza la revisión de que este sea del tipo correspondiente, es decir, si el usuario es de tipo Doctor o Asistente, el usuario a afiliar sea del tipo Paciente y viceversa.

Para añadir esta información, en ambos perfiles debe revisar si el usuario tiene afiliado a algún usuario con anterioridad y de ser así si este se encuentra en la lista del usuario, revisadas estas dos, se realiza la actualización mediante el *ViewModel* de los Usuarios.

```
fun searchUsers() {
    var counter=0
    userModel.users.observe( owner: this) { it: List<User>!
        it.forEach { it: User
            val user = it
            if (user.rutUser == searchRut) {
                if (user.typeUser != usertype) {
                    if (usertype == "Doctor" || usertype == "Familiar/Enfermero") {
                        if (user.typeUser == "Paciente") {
                            //doc o enfermero paciente
                            binding.layout.visibility = View.VISIBLE
                            binding.userMail.setText(user.email)
                            binding.userName.setText(user.fullname)
                            binding.userType.setText(user.typeUser)
                            binding.buttonAfiliar.visibility = View.VISIBLE
                            userToAfiliad = user
                        } else {
                            Toast.makeText(
                                applicationContext,
                                text: "El usuario encontrado no es compatible",
                                Toast.LENGTH_SHORT
                            ).show()
                        }
                    } else {
                        //paciente doc o enfermero
                        binding.layout.visibility = View.VISIBLE
                        binding.userMail.setText(user.email)
                        binding.userName.setText(user.fullname)
                        binding.userType.setText(user.typeUser)
                        binding.buttonAfiliar.visibility = View.VISIBLE
                        userToAfiliad = user
                    }
                } else {
                    Toast.makeText(
                        applicationContext,
                        text: "El usuario encontrado no es compatible",
                        Toast.LENGTH_SHORT
                    ).show()
                }
            }
        }
    }
}
```

Figura 56: Código buscar usuarios para afiliar, Elaboración Propia

4.11.2 Ver Registros

Los registros utilizan *RecyclerView*, que es una manera de crear listas dinámicas, en ellas se proporcionan los datos y el aspecto de cada elemento, y la biblioteca *RecyclerView* creará los elementos de forma dinámica cuando se los necesite.

```

{
    val mainActivity = this
    viewModel.meditationFlow.observe( owner: this){ it: List<Meditation>!
        binding.taskRecyclerView.apply { this: RecyclerView
            layoutManager = LinearLayoutManager(applicationContext)
            adapter = ItemViewAdapter(it, mainActivity)
            meditations = it
        }
    }
}

```

En esta sección además se implementan la función de *Swipe* y el *SwipeHelper*, que permiten que al deslizar un *RecyclerView* hacia la derecha o izquierda para eliminar o editar ese dato.

Figura 57: Código RecyclerView, Elaboración Propia

```

val swipeHelper = object : SwipeHelper( context: this){
    override fun onSwiped(viewHolder: RecyclerView.ViewHolder, direction: Int) {
        when(direction){
            ItemTouchHelper.LEFT->{
                val position = viewHolder.adapterPosition
                val builder = AlertDialog.Builder( context: this@Data_Tables)
                builder.setMessage("Seguro que quieres eliminar la medicion?")
                builder.setPositiveButton( text: "Si") { dialog, which ->
                    viewModel.deleteMeditation(meditations[position])
                    deleteRecordFirebase(meditations[position].idFirebase)
                    recyclerView.adapter?.notifyItemRemoved(position)
                }
                builder.setNegativeButton( text: "No")
                { dialog, which ->
                    recyclerView.adapter?.notifyItemChanged(position)
                }
                val alertDialog = builder.create()
                alertDialog.window?.setGravity(Gravity.BOTTOM)
                alertDialog.show()
            }
            ItemTouchHelper.RIGHT->{
                val position = viewHolder.adapterPosition
                editTaskItem(meditations[position])
            }
        }
    }
}

val itemTouchHelper = ItemTouchHelper(swipeHelper)
itemTouchHelper.attachToRecyclerView(recyclerView)

```

Figura 58: Código SwipeHelper, Elaboración Propia

4.12 Implementación Base de Datos

```

@Dao
interface MeditionDao
{
    @Query("SELECT * FROM data ORDER BY id DESC")
    fun allMeditions(): Flow<List<Medition>>

    @Query("SELECT * FROM data WHERE rutUser LIKE :user ORDER BY id DESC")
    fun allMeditionsByRutUser(user: String): List<Medition>

    @Query("SELECT * FROM data WHERE rutUser LIKE :user ORDER BY id DESC")
    fun allMeditionsByRutUserFlow(user: String): Flow<List<Medition>>

    @Insert(onConflict = OnConflictStrategy.REPLACE)
    suspend fun insertMedition(medition: Medition)

    @Update
    suspend fun updateMedition(medition: Medition)

    @Delete
    suspend fun deleteMedition(medition: Medition)
}

```

Figura 59: Código Dao Medición, Elaboración Propia

La implementación de la base de datos local en la aplicación es por medio de *Room*, por lo que se debe utilizar el modelo de DAO para el manejo de datos, por ello se utiliza un archivo para manejar la creación de la BD, otro con los métodos del CRUD, un *ViewModel* que funcionará como comunicador entre aplicación y Base de datos, finalmente se tiene el objeto para medición con sus respectivas variables que forman las tablas y otro objeto para el usuario.

En el caso de usuario ya que tiene la capacidad de tener asociada una foto de perfil, esta es transformada desde un *Bitmap* a un *ByteArray* para poder ser almacenada y viceversa para mostrarse al usuario.

```

class Converters {

    @TypeConverter
    fun fromBitmap(bitmap: Bitmap?): ByteArray {
        val outputStream = ByteArrayOutputStream()
        if (bitmap != null) {
            bitmap.compress(Bitmap.CompressFormat.PNG, quality: 100, outputStream)
        }
        return outputStream.toByteArray()
    }

    @TypeConverter
    fun toBitmap(byteArray: ByteArray): Bitmap? {
        return BitmapFactory.decodeByteArray(byteArray, offset: 0, byteArray.size)
    }
}

```

Figura 60: Código Conversión, Elaboración Propia

```

userRut = binding.textRut.text.toString()
helperRut = binding.textRutHelper.text.toString()
if (!stringCheck(helperRut)) {
    helperRut = "-"
}
if (stringCheck(userRut)) {
    when (type) {
        0 -> {
            val value1 = binding.textValue1.text.toString().toDouble()
            val value2 = binding.textValue2.text.toString().toDouble()
            val value3 = binding.textValue3.text.toString().toDouble()
            if (stringCheck(value1.toString()) && stringCheck(value2.toString()) && stringCheck(value3.toString())) {
                val newMedition = Medition(
                    userRut,
                    helperRut,
                    binding.textDate.text.toString(),
                    type,
                    value1.toString(),
                    value2.toString(),
                    value3.toString(),
                    dbId
                )
                meditionViewModel.addMedition(newMedition)
                dbRef.child(dbId).setValue(newMedition)
                    .addOnCompleteListener { it: Task<Void> ->
                        Toast.makeText(applicationContext, text: "Medicion ingresada", Toast.LENGTH_SHORT).show()
                    }.addOnFailureListener { it: Exception ->
                        Toast.makeText(applicationContext, text: "Error en Firebase", Toast.LENGTH_SHORT).show()
                    }
            }
        }
    }
}

```

Figura 61: Código añadir medición, Elaboración Propia

4.13 Implementación Reconocimiento de Imágenes

La implementación del código para el reconocimiento de imágenes utiliza herramientas de la librería de OpenCV para el procesamiento de imágenes y los pasos a seguir son los siguientes.

Para comenzar se debe ingresar una imagen de la pantalla del instrumento de medición, al usuario se le entregará herramientas de recorte de imágenes para facilitar el correcto enfoque de los números a reconocer.

La imagen se transforma a una escala de grises, pasando por una función de *threshold* para obtener las regiones oscuras de la imagen sobre el fondo iluminado de la pantalla, posteriormente se aplicará un difuminado gaussiano, y finalmente las operaciones morfológicas de dilatación para aumentar el tamaño de los objetos, seguido de apertura, que eliminará parte del ruido causado por la operación de dilatación.

Se pasarán los contornos obtenidos por una serie de filtros para poder conseguir los números o partes de números que están en los mismos, comenzando con un filtro de tamaño, que eliminará los objetos demasiado grandes o pequeños para que correspondan a un número, seguido de un segundo filtro de tamaño que calcula las dimensiones medias de los objetos de la imagen para mantener solo los que estén en el rango. Finalmente, se retiran todos aquellos contornos que correspondan a partes de números que estén separados mediante una búsqueda exterior.

La búsqueda exterior se basa en realizar conexiones a los objetos cercanos que cada contorno pueda tener y agregar estas a una lista. Cada una de estas corresponde a un posible número, que se debe revisar mediante sus coordenadas y tamaños, si la estructura que forman simboliza algún número, estas se retirarán de la lista de contornos y se agregará el número correspondiente al resultado final.

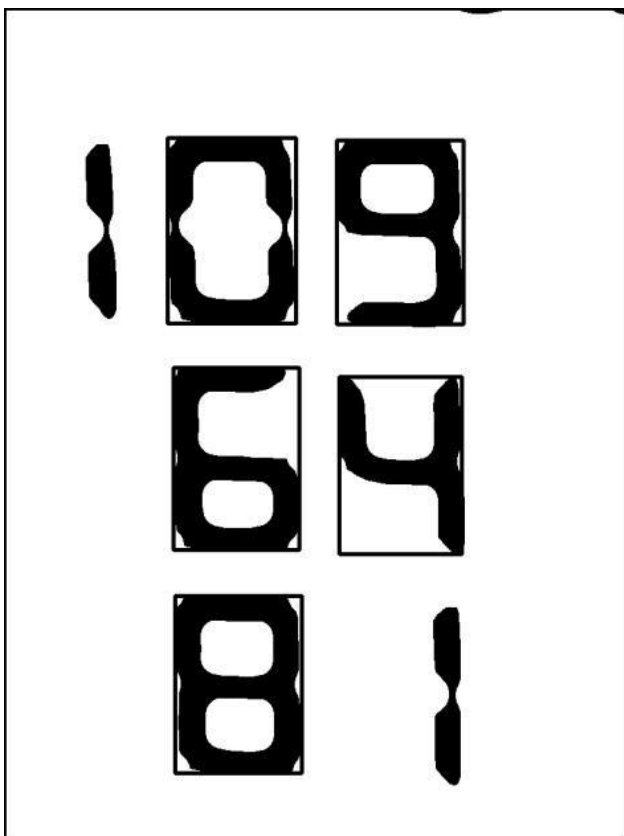


Figura 62: Procedimiento OpenCV marcado de números, Elaboración Propia

En la Figura 62 se puede ver los números marcados por el procedimiento, notar que todos los números serán marcados menos el número 1, esto se debe a que el uno es el número con menor ancho a diferencia del resto de números que comparten el mismo ancho, esto se debe a que los números que comúnmente se observan en este tipo de dispositivo corresponden a números digitales, formados por 7 líneas, que al estar todas encendidas forman un 8, si están todas encendidas menos la central se tendría un 0, de esta forma se pueden recrear todos los números del 0 al 9 prendiendo y apagando ciertas de estas líneas (Figura 63), siendo el 1 el único que no tiene ninguna línea lateral encendida.

Con los contornos de los números marcados debemos ahora comenzar el reconocimiento de cada uno, durante estos procesos se deben guardar ciertos datos que serán útiles más adelante, como las coordenadas de cada uno de los números, y en caso de encontrar un contorno cuyo grosor sea menor al establecido, se almacenará directamente un 1 en la coordenada.

El método que se utilizará en cada número será pasar un bloque a través de cada uno de los segmentos que conforman un número digital, mediante la función, *countNonZero* podemos saber en ese bloque específico cuántos de los *leds* están prendidos, si más del 50% de

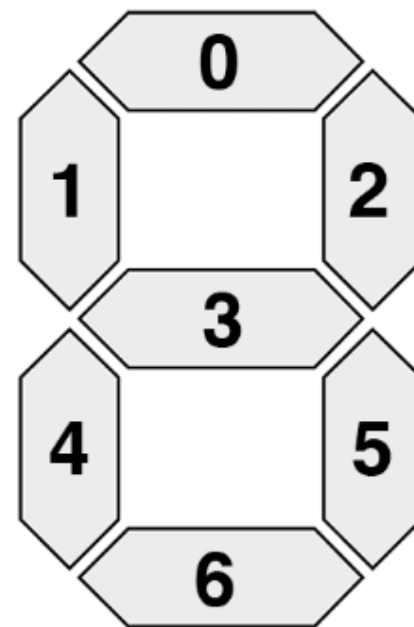


Figura 63: Número de siete segmentos, https://en.wikipedia.org/wiki/File:7_segment_display_labeled.svg

estos lo están, se puede afirmar que esa sección está prendida, así con cada sección, obteniendo una lista de ceros y unos, ya que cada una de estas lista codifica un número mediante una comparación se puede saber de qué número se está observando(Figura 64).

```
private val zero = listOf(1,1,1,0,1,1,1)
private val two = listOf(1,0,1,1,1,1,0)
private val three= listOf(1,0,1,1,0,1,1)
private val four = listOf(0,1,1,1,0,1,0)
private val five = listOf(1,1,0,1,0,1,1)
private val six = listOf(1,1,0,1,1,1,1)
private val seven= listOf(1,0,1,0,0,1,0)
private val eight= listOf(1,1,1,1,1,1,1)
private val nine = listOf(1,1,1,1,0,1,1)
```

Figura 64: Lista códigos de cada número, Elaboración Propia

Finalmente, cada número junto con su coordenada se pasarán por la siguiente función (Figura 65) que según su ubicación en el eje x ordenará los números, a menos que la diferencia entre el eje y sea muy grande, en este caso, corresponde a otro número. De esta manera se obtendrá cada uno en orden.

Los valores encontrados se enviarán al formulario para este los ingrese de manera automática y el usuario puede revisarlos antes de subirlos a la base de datos de igual manera que el resto de las mediciones.

```

private fun processNumbers(){
    var firstindex = finalnumbers.indexOfFirst{ it.coorNum == 9999 }
    var index = finalnumbers.indexOfFirst{ it.coorNum == 9999 }
    while (index > -1){
        if(index==0){
            finalnumbers.removeAt(index)
            index = finalnumbers.indexOfFirst{ it.coorNum == 9999 }
        }
        else{
            var list = finalnumbers.subList(firstindex, index)
            stringOfNums.add(orderNumbers(list))
            firstindex = finalnumbers.indexOfFirst{ it.coorNum == 9999 }
            finalnumbers.removeAt(index)
            index = finalnumbers.indexOfFirst{ it.coorNum == 9999 }
            if (index == -1 ){
                list = finalnumbers.subList(firstindex,finalnumbers.size)
                stringOfNums.add(orderNumbers(list))
            }
        }
    }
}

private fun orderNumbers(list: MutableList<Coor>):String{
    var num:String = ""
    list.sortBy { it.coorNum }
    list.forEach { num += it.valueNum.toString()}
    Log.e( tag: "TAG", msg: "$num")
    return num
}

```

Figura 65: Código Orden de números, Elaboración Propia

```

private fun convertImageBnW(bitmap: Bitmap): Bitmap {
    val bitmapAux: Bitmap = bitmap.copy(Bitmap.Config.ARGB_8888, isMutable: true)
    val src:Mat = Mat(bitmap.height,bitmap.width,CvType.CV_8UC1)
    val dest:Mat = Mat(bitmap.height,bitmap.width,CvType.CV_8UC1)
    Utils.bitmapToMat(bitmapAux,src)
    Imgproc.cvtColor(src,dest, Imgproc.COLOR_BGR2GRAY)
    val processImg: Bitmap = Bitmap.createBitmap(dest.cols(),dest.rows(),Bitmap.Config.ARGB_8888)
    Utils.matToBitmap(dest,processImg)
    return processImg
}

private fun convertImageGaussian(bitmap: Bitmap): Bitmap {
    val bitmapAux: Bitmap = bitmap.copy(Bitmap.Config.ARGB_8888, isMutable: true)
    val src:Mat = Mat(bitmap.height,bitmap.width,CvType.CV_8UC4)
    val dest:Mat = Mat(bitmap.height,bitmap.width,CvType.CV_8UC4)
    Utils.bitmapToMat(bitmapAux,src)
    //Imgproc.medianBlur(src,dest,5)
    Imgproc.GaussianBlur(src,dest, Size( width: 3.0, height: 3.0), sigmaX: 0.0)
    val processImg: Bitmap = Bitmap.createBitmap(dest.cols(),dest.rows(),Bitmap.Config.ARGB_8888)
    Utils.matToBitmap(dest,processImg)
    return processImg
}

```

Figura 66: Código algunas transformaciones de la imagen, Elaboración Propia

4.14 Implementación de *Dashboard*

La función fundamental del dashboard para usuarios que tengan pacientes asociados es poder tener una lista de los mismos que se pueda cambiar con facilidad, para encontrar estos usuarios y obtener su información se utilizan las siguientes funciones (Figura 67), *getUsers* se encarga de crear una lista con todos los tus asociados a este usuario, cada uno de los nombres encontrados entran en la lista que sirve como selector para el usuario, al momento que este realiza una selección, la información de este se actualizará en la interfaz, por otra lado la función *getUserById* cumple la tarea de obtener los datos del usuario al que pertenece la sesión y agregar sus datos personales en el menú lateral.

```

fun getUserById(userId: Int) {
    lifecycleScope.launch(Dispatchers.IO){ this: CoroutineScope
        user = userModel.getUserById(userId)
        Log.e( tag: "asd", user.toString())
    }.invokeOnCompletion { it: Throwable?
        this.runOnUiThread(java.lang.Runnable {
            binding.nameText.text = user.fullname
            navHeader.findViewById<TextView>(R.id.userName).setText(user.fullname)
            navHeader.findViewById<TextView>(R.id.userMail).setText(user.email)
            navHeader.findViewById<TextView>(R.id.userType).setText(user.typeUser)
            navHeader.findViewById<ImageView>(R.id.imgProfile).load(user.profilePic)
            binding.imgProfile.load(user.profilePic)
            //revisar
        })
    }
}

fun getUsers(afiliatedRut:String): MutableList<String>{
    ruts = afiliatedRut.split( ...delimiters: ";" )
    val names:MutableList<String> = mutableListOf()
    userModel.users.observe( owner: this){ it: List<User>!
        it.forEach{ it: User
            val users = it
            ruts.forEach { it: String
                if (it == users.rutUser) {
                    names.add(users.fullname)
                }
            }
        }
    }
    return names
}

```

Figura 67: Código *GetUsers UserbyId*, Elaboración Propia

4.15 Implementación de comunicación Bluetooth

Android proporciona una API Bluetooth para realizar diferentes operaciones, para acceder se utilizará un *BluetoothAdapter*, que proporciona el servicio *Bluetooth* de *Android*, con este se podrá saber si el usuario tiene o no activo el servicio de BT y pedir su habilitación en caso de que esté apagado.

Con el servicio revisado y encendido, se debe realizar la búsqueda de dispositivos enlazados, en el caso de las ESP32 con las que se comunicarán se añadirán previamente mediante la configuración *bluetooth* del dispositivo móvil, para que aparezcan en esta búsqueda y el usuario pueda seleccionarlas, mediante el botón de conectar que establecerá la conexión y notificará al usuario.

Finalmente se puede iniciar la medición, esto se le comunicará a la ESP32 mediante el envío de un comando, esta lo recibirá y empezará a enviar los datos hacia el dispositivo móvil. Una *Thread* se creará para manejar el recibo de datos desde el fondo, de esta forma el usuario podrá ver sus datos además de detener la medición o guardar los resultados obtenidos.

Para el caso del ECG se tiene además una gráfica para mostrar los datos, que se actualizará a tiempo real, y al guardar los datos se creará un archivo con todos los valores, para que el usuario pueda realizar un *ploteo* de la información si así lo requiere.

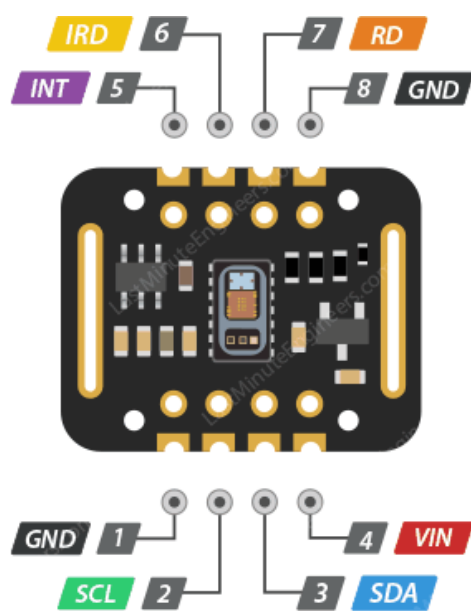
Por el lado de la ESP32 se utiliza la librería de *BluetoothSerial* la que permitirá la comunicación con la aplicación, se le da un nombre al dispositivo para que se pueda encontrar sin necesidad de saber la dirección MAC, realizada la conexión la ESP32 espera a recibir un determinado mensaje para comenzar las mediciones *readData()*, la cuales serán enviadas a la aplicación a medida que estas se obtengan mediante la función *sendData()*.

```
private inner class ConnectedThread() : Thread() {
    override fun run() {
        val inputStream = m_bluetoothSocket!!.inputStream
        val buffer = ByteArray(1024)
        var bytes = 0
        counter = 0;

        while (startBT) {
            try {
                bytes = inputStream.read(buffer, bytes, 1024 - bytes)
                val receivedMessage = String(buffer).substring(0, bytes)
                Log.e(tag, "asd", msg: "New received message: " + receivedMessage)
                if(receivedMessage.get(0).isDigit() && counter == 0){
                    val message = receivedMessage.split(Regex(" "))
                    if (message[1].toInt() > 40 && message[1].toFloat() < 200 && message[0].toFloat() > 90 && message[0].toFloat() < 100){
                        sp02.add(message[0].toFloat())
                        hb.add(message[1].toInt())
                        val mess1 = "Sp02: " + message[0] + " HR: " + message[1]
                        val mess2 = "Avg Sp02: " + (round(x: sp02.average()*100)/100).toString() + " Avg HR: " + (round(x: hb.average()*100)/100).toString()
                        runOnUiThread {
                            binding.textView3.text = mess1
                            binding.textView4.text = mess2
                        }
                    }
                }else{
                    if(receivedMessage.get(0).isDigit()){
                        addEntry(receivedMessage.toInt())
                    }
                }
            }
            if(receivedMessage == "ECG"){
                counter=1
            }
            //binding.textView3.text = receivedMessage
            bytes = 0
        } catch (e :IOException) {
            e.printStackTrace()
            Log.d(tag, "asd", msg: "Error reading")
            break
        }
    }
}
```

Figura 68: Código *Thread* receptora de datos, Elaboración Propia

4.16 Implementación Max30102



MAX30102 Module Pinout

Figura 69: Max30102 Pins Layout,
<https://lstminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/>



- VIN es el pin de alimentación. Puedes conectarlo a la salida de 3.3V o 5V de la ESP32.
- SCL es el pin de reloj I2C, conéctalo a la línea de reloj I2C de la ESP32.
- SDA es el pin de datos I2C, conéctalo a la línea de datos I2C de la ESP32.
- GND corresponde a tierra.

Estos son los *pines* relevantes para el proyecto y los que fueron conectados directamente a la ESP32.

Librería *SparkFun* MAX3010x da acceso a una serie de funciones que son clave para obtener datos del biosensor, sin embargo, esta librería

no da resultados consistentes, es posible conseguir algunas lecturas correctas pero gran parte de estas corresponderán a errores con valores de -999.

Debido a esto se utilizará la librería de *Maxim Integrated, Inc.*, que contiene las mismas funciones encontradas en *SparkFun*, pero da mejores resultados, además de utilizar menos memoria, permitiendo que todo el sistema funcione en la ESP32, recibo y envié de data con el dispositivo móvil, así como la comunicación con la pantalla LCD para mostrar los valores obtenidos.

Para obtener los valores de oxígeno en la sangre y ritmo cardiaco se utiliza la función *rf_heart_rate_and_oxygen_saturation*, la que funciona de manera similar a *SparkFun*, obteniendo los valores de la luz Roja e IR, sin embargo, estos se obtienen mediante las funciones *Wire* de *arduino* y pasan por un algoritmo de corrección.

Los valores serán enviados a la aplicación donde se realizará un promedio de los datos obtenidos que se le mostrará al paciente, cuando él quiera puede detener la medición y guardar sus datos en una medición.

```

void loop() {
  float n_spo2, ratio, correl; //SPO2 value
  int8_t ch_spo2_valid; //indicator to show if the SPO2 calculation is valid
  int32_t n_heart_rate; //heart rate value
  int8_t ch_hr_valid; //indicator to show if the heart rate calculation is valid
  int32_t i;
  char hr_str[10];

  //buffer length of BUFFER_SIZE stores ST seconds of samples running at FS sps
  //read BUFFER_SIZE samples, and determine the signal range
  for(i=0;i<BUFFER_SIZE;i++)
  {
    while(digitalRead(oxiInt)==1); //wait until the interrupt pin asserts
    maxim_max30102_read_fifo((aun_red_buffer+i), (aun_ir_buffer+i)); //read from MAX30102 FIFO
#ifdef DEBUG
    Serial.print(i, DEC);
    Serial.print(F("\t"));
    Serial.print(aun_red_buffer[i], DEC);
    Serial.print(F("\t"));
    Serial.print(aun_ir_buffer[i], DEC);
    Serial.println("");
#endif // DEBUG
  }

  //calculate heart rate and SpO2 after BUFFER_SIZE samples (ST seconds of samples) using Robert's method
  rf_heart_rate_and_oxygen_saturation(aun_ir_buffer, BUFFER_SIZE, aun_red_buffer, &n_spo2, &ch_spo2_valid, &n_heart_rate, &ch_hr_valid, &ratio, &correl);
}

```

Figura 70: Parte del Código ESP32 Max30102, Elaboración Propia

4.17 Implementación ADB232

Según la Figura 71, se conecta 3.3v y GND del sensor a 3.3v y GND del ESP32 respectivamente, el pin OUTPUT del sensor a VP(SP)(A0)(pin5) del ESP32 y finalmente LO- y LO+ están conectados al pin40(RX0) y pin41(TX0) respectivamente.

Para obtener los datos del biosensor, se debe configurar los pines 41 y 40 en INPUT para que iniciar la detección de LO- y LO+, se debe revisar que ambos lectores digitales estén funcionando, antes de obtener los valores que entrega el lector análogo VP y estos puedan ser enviados por medio de *bluetooth* al dispositivo móvil.

Por el lado de la aplicación mediante la librería *MPAndroidChart* que permite realizar una gráfica de línea que se actualiza en vivo, mediante la misma *Thread* que se usa para recibir y actualizar los valores del Max30102, de esta forma el usuario puede ver su ECG en vivo mediante la aplicación.

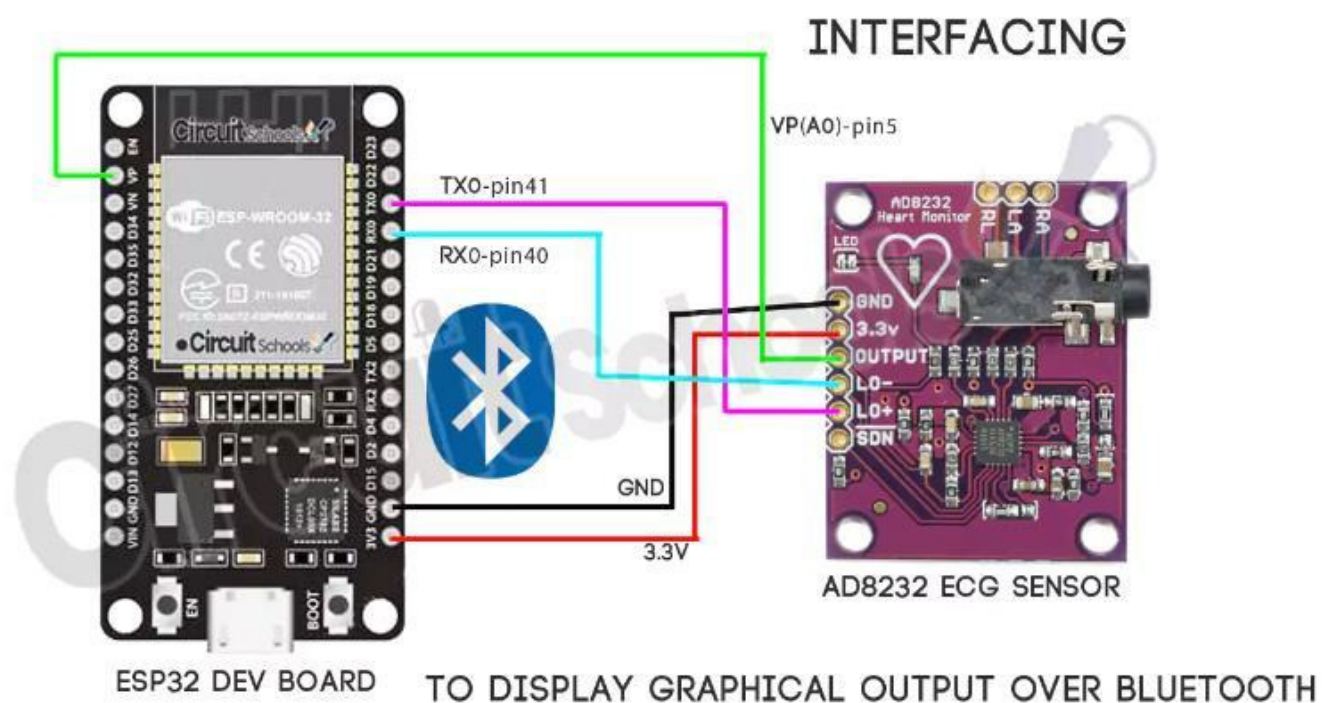


Figura 71: Conexiones con la ESP32, <https://www.circuitschools.com/ecg-monitoring-system-using-ad8232-with-arduino-or-esp32-iot-based/>

```

void readData(){
  if (ESP_BT.available())
  {
    income = ESP_BT.read();
    if (income == "B"){
      ESP_BT.write(1);
    }else{
      income="A";
    }
  }
}

void sendData(int32_t dataecg){
  String sending = String(dataecg);
  ESP_BT.print(sending);
}

void setup() {
  // initialize the serial communication:
  Serial.begin(9600);
  ESP_BT.begin("ESP32_ECG");
  Serial.println(); // blank line in serial ...
  pinMode(41, INPUT); // Setup for leads off detection LO +
  pinMode(40, INPUT); // Setup for leads off detection LO -
  // initialize the serial BT communication:
  income = "R";
  while(income != "A"){
    Serial.println(F("Waiting.."));
    if (ESP_BT.available()) {
      income = ESP_BT.read();
      if (income == "A"){
        ESP_BT.print("ECG");
      }
    }
  }
  delay(1000);
}
}

```

Figura 73: Código ADB232 Parte 1, Elaboración Propia

```

void loop() {

if((digitalRead(40) == 1)|| (digitalRead(41) == 1)){
  Serial.println('!');
  //ESP_BT.println('!');
}
else{
  readData();
  if (income == "A"){
    // send the value of analog input 0:
    Serial.println(analogRead(A0));
    sendData(analogRead(A0));
  }else{
    while(income != "A"){
      Serial.println(F("Waiting.."));
      if (ESP_BT.available()) {
        income = ESP_BT.read();
        if (income == "A"){
          ESP_BT.println("Start ECG");
        }
      }
    }
    delay(1000);
  }

}

//Wait for a bit to keep serial data from saturating
delay(5);
}
}

```

Figura 72: Código ADB232 Parte 2, Elaboración Propia

CAPÍTULO 5: CONCLUSIONES

Durante este trabajo se ha desarrollado un sistema que integra múltiples tecnologías con el propósito de cumplir los objetivos propuestos e intentar llegar a una solución que sea capaz de facilitar el monitoreo de pacientes, tanto de ellos mismos como a funcionarios del área de la salud. El sistema combina tecnologías como biosensores, micro controladores, comunicación bluetooth, desarrollo de aplicaciones, reconocimiento de imágenes y manejo de datos.

Aprovechando estas tecnologías, el sistema es capaz de recopilar las constantes vitales del usuario por medio de distintos métodos de forma que pueda escoger la más conveniente para su situación particular, y comunicar el estado de cada paciente a los funcionarios de la salud que estén afiliados, generando de esta forma una red de comunicación expedita y conveniente para ambas partes involucradas.

El principal objetivo que se propuso, fue crear un sistema capaz de integrar distintos aparatos de monitoreo médico simples, esto se logró gracias a los biosensores que se encuentran en el mercado de los micro controladores, como el biosensor MAX30102 o AD8232. Estos dispositivos acercan tecnología de sensores de vigilancia de la salud, por medio de librerías y fácil acceso comercial, en este apartado cabe destacar la existencia de muchos otros de este tipo de módulos que se estudiaron, algunos ejemplos son ADA-1093 sensor de pulso, Sensor GY-273 QMC5883L magnetómetro, la placa *MySignals* HW y *E-Health Sensor Shield V2.0*, estas contienen mediciones de todas las constantes vitales además de EMG, GSR y glucómetro, sin embargo, su valor es muy elevado por lo que pocos usuarios tendrían acceso a estas tecnologías, requieren conocimientos elevados en el área de la salud para poder utilizarse o son de acceso limitado debido a que su producción se discontinuó.

Los datos obtenidos se puedan almacenar y enviar a personal capacitado para realizar revisiones periódicas remotas, esto se logra con la combinación de base de datos *Firebase* y *Room*, permitiendo que la aplicación funcione tanto con o sin conexión, y dándole acceso al usuario a sus mediciones desde cualquier dispositivo al sincronizar los datos con la nube, junto con el sistema de afiliados para que se pueda compartir la información a los profesionales de la salud que sea conveniente.

Finalmente, generar alertas en el caso de encontrar anomalías en dichas mediciones. Para esto el profesional de salud asignado al paciente tiene acceso a las mediciones del paciente, para poder dar un estimado del estado del paciente, que será notificado con una simbología de colores en su *dashboard*, de esta forma tanto el usuario como su asistente sabrán el estado y podrán actuar de manera preventiva.

Si bien se lograron los objetivos propuestos, durante el desarrollo se encontraron algunas limitaciones, como el acceso a los signos vitales del paciente, la cual se abordó permitiendo que el ingreso de mediciones de manera manual y por foto, sin embargo, con

fácil acceso a estas se puede realizar un proceso más automatizado para el usuario. Otra limitación que cabe destacar es el manejo de memoria RAM en microcontroladores, ya que al ahora de utilizar múltiples biosensores en una sola placa junto con una pantalla y menú para mejorar la experiencia del usuario, resulta fácil sobrepasar los límites que controladores como el microcomputador Arduino Nano o ESP32 tienen, provocando que el sistema no sea operable. Esta limitación en particular se abordó separando los biosensores, de esta manera cada microcontrolador puede concentrarse en su tarea particular, además al realizar una conexión con el dispositivo móvil, permite que este último tome la carga de guardar, ordenar y trabajar los datos, liberando aún más las labores de los micro controladores.

Los dos mayores desafíos que se tuvieron que resolver para llegar al sistema final fueron, el manejo de datos para que la aplicación funcionará sin conexión pero a su vez fuera capaz de enviar los datos al profesional de la salud y este tuviese acceso a los datos de sus pacientes asignados, para lo que *Firebase* resultó ser la herramienta ideal, ya que simplificó el manejo de solicitudes que se tenía en un principio, que, pese a resolver la problemática principal, almacenar los datos del paciente para posteriormente presentárselos a los afiliados del usuario, resultaba en extensas esperas para que los datos fueran enviados y recibidos por *MySQLCloud* además que se debían manejar los momentos que donde se perdía por completo la conexión de manera que estos datos no se desvanecieran.

El segundo desafío a resolver fue la lectura de números en pantalla, que resultó ser un problema, debido a que los dígitos que muestran los *leds* de los sensores médicos no son comprendidos por herramientas de reconocimiento de texto, ya que la separación que cada parte del número no permite una buena lectura, las herramientas de Open CV y manejo de imágenes permitieron generar una solución a la hora de identificar estos números con un grado mucho mayor de precisión.

Por otro lado, uno de los mayores aportes que este tipo de sistema puede contribuir se encuentra en la implementación en zonas rurales, donde permitiría mantener un seguimiento de muchos pacientes con una sola persona que recopile las mediciones del área y suba los datos recopilados para que el profesional de la salud pueda ver los estados generales de estas personas, optimizando los tiempos de consulta del médico y permitiendo que se enfoque principalmente en los pacientes que vea que tengan un estado más crítico.

Otra problemática que presenta la salud en sectores rurales son las constantes rotaciones de médicos y profesionales que tienen estas áreas, teniendo un acceso a todas las mediciones del paciente a través del tiempo se puede reducir un poco esta problemática, permitiendo a cada uno de los profesionales que estén en la rotación ver toda la información histórica de ese particular paciente.

Además, permitiría el monitoreo no sólo de profesionales de la salud, sino que familiares que no tienen los medios o el tiempo para visitar a ese paciente que se encuentra en un

área rural o apartada, puedan conocer el estado de su familiar a la distancia y estar alertas ante alguna emergencia.

Este trabajo ha analizado los beneficios y los retos asociados a la implantación de un sistema así y tratado de aportar ideas para los profesionales de la salud y los responsables políticos que buscan mejorar la seguridad de los pacientes y la calidad de la atención.

La investigación realizada indica que un sistema integrado de monitorización de pacientes puede mejorar la toma de decisiones clínicas y reducir los resultados adversos al permitir la detección precoz del deterioro del paciente. Sin embargo, la implantación de esta tecnología también presenta retos, como la gestión de los datos, las limitaciones técnicas y la necesidad de formación y educación del personal.

Para superar estos retos, las organizaciones de la salud deben dar prioridad a la selección de las tecnologías adecuadas, establecer políticas claras de gestión de datos y proporcionar formación y educación suficientes a los miembros del personal. Los resultados de este trabajo sugieren que los beneficios de implantar un sistema integrado de monitorización de pacientes superan a los retos y que el éxito de la implantación requiere un esfuerzo de colaboración entre los profesionales de la salud y los proveedores de tecnología.

Las proyecciones y desafíos a superar para este prototipo sería realizar pruebas en entornos reales, como en sectores rurales, hospitales o centros médicos, de esta forma se podría consolidar el funcionamiento y encontrar mejoras o fallos, permitiendo que el sistema se perfeccione. Por otro lado un punto de mejora del sistema, que no requeriría de este tipo de pruebas, se encuentra en la maneras y opciones a la hora de ingresar datos, agregar más biosensores que se pudieran conectar de manera directa con la aplicación facilitaría al usuario el ingreso de datos, asimismo agregar más instrumentos de medición que el sistema pueda leer le daría al usuario más opciones a la hora de ingresar sus datos.

En conclusión, el sistema integrado de monitorización de pacientes representa una importante oportunidad para mejorar la seguridad y los resultados de los pacientes. Las organizaciones médicas que implanten esta tecnología de forma eficaz pueden mejorar la calidad de la atención prestada a los pacientes y contribuir al avance de la asistencia médica. Con este proyecto se buscó ofrecer valiosas ideas y recomendaciones, para quien desee implantar un sistema integrado de monitorización de pacientes, se capaz de alcanzar sus objetivos.

REFERENCIAS BIBLIOGRÁFICAS

- ¿Qué es el Internet de las cosas (IoT)? (s.f.). Obtenido de RedHat:
<https://www.redhat.com/es/topics/internet-of-things/what-is-iot>
- Adebayo, K. S. (28 de December de 2022). *6 healthcare AI predictions for 2023*. Obtenido de VentureBeat: <https://venturebeat.com/ai/6-healthcare-ai-predictions-for-2023/>
- Afzal, S. (02 de Septiembre de 2016). *Analog Devices*. Obtenido de I2C Primer: What is I2C? (Part 1): <https://www.analog.com/en/technical-articles/i2c-primer-what-is-i2c-part-1.html>
- Android Studio. (07 de Febrero de 2023). *Introducción a Android Studio*. Obtenido de Developers Android: <https://developer.android.com/studio/intro?hl=es-419>
- Arduino. (5 de Febrero de 2018). Obtenido de What is Arduino?:
<https://www.arduino.cc/en/Guide/Introduction>
- Asistenciales, S. d. (2018). *Programa Nacional de Telesalud*. Ministerio de Salud.
- Awati, R. (Mayo de 2022). *TechTarget*. Obtenido de EEPROM (electrically erasable programmable read-only memory): <https://www.techtarget.com/whatis/definition/EEPROM-electrically-erasable-programmable-read-only-memory>
- Barros, L. (3 de enero de 2015). *¿ES LO MISMO TELESALUD QUE TELEMEDICINA?* Obtenido de Doctor a un clic: <https://www.doctoraunclitic.com/blog/es-lo-mismo-que-telesalud-que-telemedicina/>
- Brousse, D. L. (2006). Telemedicina. Una herramienta poco explotada. *REVISTA DE OBSTETRICIA Y GINECOLOGÍA HOSPITAL SANTIAGO ORIENTE*, 233, Pagina 16.
- C, C. (19 de Marzo de 2020). *Physiology, Respiratory Rate In: StatPearls*. Obtenido de NCBI: <https://www.ncbi.nlm.nih.gov/books/NBK537306/>
- CardiacSense. (s.f.). *Monitorizacion Remota del Paciente*. Obtenido de CardiacSense: <https://www.cardiacsense.com/monitorizacion-remota-del-paciente/>
- CENS. (2020). *Guía de Buenas Prácticas y Recomendaciones*. Santiago: CENS (Centro Nacional en Sistemas de Informacion de Salud).
- CuidadoDeSalud.gov. (s.f.). *Glosario: CuidadoDeSalud.gov*. Obtenido de CuidadoDeSalud.gov: <https://www.cuidadodesalud.gov/es/glossary/>
- Digital Guide IONOS. (01 de Diciembre de 2022). *¿Qué es Bluetooth? Toda la información sobre el estándar inalámbrico*. Obtenido de Digital Guide IONOS: <https://www.ionos.es/digitalguide/servidores/know-how/que-es-bluetooth/>
- Finnegan, G. (13 de Nov de 2020). *Is MedTech ready for patient engagement?* . Obtenido de PFMD.org: <https://patientfocusedmedicine.org/is-medtech-ready-for-patient-engagement/>
- Firebase. (08 de Agosto de 2022). *Firebase Realtime Database*. Obtenido de Firebase: <https://firebase.google.com/docs/database?hl=es-419>

- Frost & Sullivan. (27 de Mayo de 2020). *Frost & Sullivan Reveals Non-contact Patient Monitoring Technologies to Revolutionize Healthcare*. Obtenido de Cision PR Newswire: <https://www.prnewswire.com/in/news-releases/frost-amp-sullivan-reveals-non-contact-patient-monitoring-technologies-to-revolutionize-healthcare-857564628.html>
- G. Wu, S. T. (Abril de 2011). *M2M: From mobile to embedded internet*. IEEE Commun. Mag., vol. 49.
- Gabay, J. (28 de Octubre de 2015). *Tecnología de sensores para aplicaciones de salud y estado físico*. Obtenido de Digi-Key: <https://www.digikey.com/es/articles/sensor-technology-for-health-and-fitness-applications>
- Ginés, E. M. (19 de Julio de 2019). *Sensores Arduino*. Obtenido de aprendiendoarduino: <https://aprendiendoarduino.wordpress.com/2018/04/14/sensores-arduino-3/>
- Girodmedical, M. D. (20 de Febrero de 2017). *Espirómetro: para qué sirve*. Obtenido de GirodMedical: https://www.girodmedical.es/blog_es/espirometro-para-que-sirve/
- Gonzalez, A. N. (09 de Febrero de 2011). *¿Qué es Android?* Obtenido de Xataka Android: <https://www.xatakandroid.com/sistema-operativo/que-es-android>
- Gracia, M. (s.f.). *IoT - Internet Of Things*. Obtenido de Deloitte: <https://www2.deloitte.com/es/es/pages/technology/articles/loT-internet-of-things.html>
- Granath, E. (01 de Abril de 2020). *Power & Beyond*. Obtenido de The applications of Op-Amp integrators: [https://www.power-and-beyond.com/the-applications-of-op-amp-integrators-a-915207/#:~:text=An%20operational%20amplifier%20\(op-amp,input%20voltage%20integrated%20over%20time.](https://www.power-and-beyond.com/the-applications-of-op-amp-integrators-a-915207/#:~:text=An%20operational%20amplifier%20(op-amp,input%20voltage%20integrated%20over%20time.)
- Hernández, L. d. (01 de Diciembre de 2021). *Visión artificial, OpenCV y Python*. Obtenido de Programar Facil: <https://programarfacil.com/podcast/81-vision-artificial-opencv-phyton/>
- Hiraoka. (13 de enero de 2022). Obtenido de Tensiómetro: ¿qué es, para qué sirve y cómo funciona?: <https://hiraoka.com.pe/blog/post/tensiometro-que-es-para-que-sirve-y-como-funciona>
- How to Electronics. (19 de Noviembre de 2022). *ECG Graph Monitoring with AD8232 ECG Sensor & Arduino*. Obtenido de How to Electronics: <https://how2electronics.com/ecg-monitoring-with-ad8232-ecg-sensor-arduino/>
- How, K. (s.f.).
- IBM. (08 de Abril de 2019). *¿Qué es la Visión Artificial?* Obtenido de IBM: <https://www.ibm.com/cloud/topics/computer-vision>
- Joseba Rabanales Sotosa, I. P.-T. (2011). Tecnologías de la Información y las Comunicaciones: Telemedicina. *REV CLÍN MED FAM*, vol. 4, pp. 42–48.
- Last Minutes Engineers. (2018, Julio). *Interfacing MAX30102 Pulse Oximeter and Heart Rate Sensor with Arduino*. Retrieved from Last Minutes Engineers: <https://lastminuteengineers.com/max30102-pulse-oximeter-heart-rate-sensor-arduino-tutorial/>

- Lawson, J. (24 de Mayo de 2021). *Sistema De Equipo De Dispositivo De Monitoreo De Pacientes Pronóstico Del Mercado, Por Países, Tipo Y Aplicación, Con Pronóstico De Ventas, Precio, Ingresos Y Tasa De Crecimiento, 2020-2026*. Obtenido de slp.news: <https://www.slpnewsmx.com/perspectivas-del-mercado-global-de-sistema-de-equipo-de-dispositivo-de-monitoreo-de-pacientes-hasta-2026/>
- Leiva, L. (2019, Mayo 13). Fonasa: solo 13% de mujeres entre 20 y 65 años usó el examen preventivo gratuito. *La Tercera*.
- Mallick, S. (14 de Noviembre de 2016). *Image Recognition and Object Detection : Part 1*. Obtenido de LearnOpenCV: <https://learnopencv.com/image-recognition-and-object-detection-part1/>
- Mansillas, J. (20 de Diciembre de 2020). CIENCIA&SALUD: IMPACTO Y DESARROLLO DE LA TELEMEDICINA EN CHILE. (V. Torres, Entrevistador) CENS (Centro Nacional en Sistemas de Información en Salud).
- Matamala, A. O. (Noviembre de 2012). *Pediatría Integral*. Obtenido de Lectura del ECG: <https://www.pediatriaintegral.es/numeros-anteriores/publicacion-2012-11/lectura-del-ecg/>
- Montini., A. V. (2009). *Tecnología de monitoreo remoto inalámbrico*. Ciudad de México: Arch. Cardiol. Méx. vol.79 supl.2.
- Mora, S. L. (17 de Mayo de 2020). *Firestore: qué es, para qué sirve, funcionalidades y ventajas*. Obtenido de Digital55: <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>
- Navarro, C. R. (02 de Mayo de 2017). *SENSORES MEDICOS ARDUINO*. Obtenido de Soloelectronicos: <https://soloelectronicos.com/tag/sensores-medicos-arduino/>
- OKDIARIO. (2019, Septiembre 14). Qué es la prevención de la salud. *Ok Diario*.
- Ortega, D. H. (21 de 12 de 2017). *hetpro-store*. Obtenido de Amplificador Operacional – qué es y sus configuraciones más usadas: <https://hetpro-store.com/TUTORIALES/amplificador-operacional/>
- RaspberryPi*. (Mayo de 2012). Obtenido de ¿Que es Raspberry Pi?: <https://raspberrypi.cl/que-es-raspberry/>
- Redón, J. G. (2012). *Manual de La Enfermería, Novena*. OCEANO/CENTRUM.
- Sanitas*. (22 de enero de 2000). Obtenido de ¿Que es la tensión arterial?: <https://www.sanitas.es/sanitas/seguros/es/particulares/biblioteca-de-salud/cardiologia/salud-cardiovascular/sin012200wr.html>
- Santos, R. H., Elles, L. Z., Julio, A. J., & Novoa, A. G. (2018). Diseño de un sistema para el monitoreo remoto de signos vitales en pacientes. *PERSPECTIV@S. Revista de Tecnología e Información*, 1-5.
- Solectro*. (12 de Junio de 2021). Obtenido de ¿Qué son los sensores analógicos? Todo sobre su funcionamiento: <https://solectroshop.com/es/blog/que-son-los-sensores-analogicos-todo-sobre-su-funcionamiento-n91>

soloelectronicos. (2 de Mayo de 2017). *soloelectronicos.com*. Obtenido de Plataforma de sensores e-Health V1.0 para Arduino y Raspberry Pi [aplicaciones biométricas / medicina]: <https://soloelectronicos.com/2017/05/02/plataforma-de-sensores-e-health-v1-0-para-arduino-y-raspberry-pi-aplicaciones-biometricas-medicina/>

Subdirección General de informática de INSALUD, Subdirección General de Sistemas y Tecnologías de la Información del Ministerio de Sanidad y Consumo. (2000). Plan de telemedicina del INSALUD. *Ministerio de Sanidad y Consumo y Dirección General de Organización y Planificación Sanitaria de INSALUD*.

Subsecretaría de Redes Asistenciales. (2018). *Programa Nacional de Telesalud*. Santiago: Ministerio de Salud.

Surabhi Pandey, S. S. (Septiembre de 2020). *Digital Therapeutics Market by Application (Diabetes, Obesity, Cardiovascular Disease (CVD), Central Nervous System (CNS) Disease, Respiratory Disease, Smoking Cessation, Gastrointestinal Disorder (GID), and Others), Product Type (Software and Device), an*. Obtenido de Avenue: <https://www.alliedmarketresearch.com/digital-therapeutics-market>

Tipos... (2016). Obtenido de 10Tipos: <http://10tipos.com/tipos-de-termometros/>

Torres, O. S., & Lugo, L. M. (2010). *Telemedicina: ¿futuro o presente?* Ciudad de La Habana: Revista habanera de ciencias médicas v.9 n.1.

Torres, V. (20 de Diciembre de 2020). Impacto y desarrollo de la telemedicina en Chile. (J. Mansillas, Entrevistador)

Tortora, G. J. (2020). *Principios de Anatomía y Fisiología (15.a ed.)*. Wiley.

Universidad de Chile . (2020). *Estación monitoreo remoto de pacientes COVID-19*. Obtenido de <http://www.dii.uchile.cl/2020/05/12/estacion-monitoreo-remoto-de-pacientes-covid-19/>: <http://www.dii.uchile.cl/2020/05/12/estacion-monitoreo-remoto-de-pacientes-covid-19/>

Vay, L. D.-V. (2019). *Anatomía y fisiología humana (2.a ed.)*. Paidotribo.

Weiss, G. &. (2019). Smartphone and Smartwatch-Based Biometrics Using Activities of Daily Living. *IEEE Access.*, PP. 1-1. 10.1109/ACCESS.2019.2940729.

Yanez, M. G. (2013). *Glucose Meter Fundamentals and*. Tempe, Arizona 85284: Freescale Semiconductor, Inc.

Yida. (30 de Diciembre de 2019). *Biomedical Sensors: Types of sensors and How it works*. Obtenido de Seeedstudio: <https://www.seeedstudio.com/blog/2019/10/14/biomedical-sensors-types-of-sensors-and-how-it-works/>

ANEXOS

En los **Anexos** se incluye todo aquel material complementario que no es parte del contenido de los capítulos de la memoria, pero que permiten a un lector contar con un contenido adjunto relacionado con el tema.