



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Politécnica Superior de Gandia

Recogida y análisis de los datos de un pulsioxímetro a  
través de una aplicación móvil.

Trabajo Fin de Grado

Grado en Tecnologías Interactivas

AUTOR/A: Vaello Pérez, Adrián

Tutor/a: Lloret Mauri, Jaime

CURSO ACADÉMICO: 2021/2022

UNIVERSITAT POLITÈCNICA DE VALÈNCIA  
ESCOLA POLITÈCNICA SUPERIOR DE GANDIA  
GRADO EN TECNOLOGIAS INTERACTIVAS

---



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



ESCOLA POLITÈCNICA  
SUPERIOR DE GANDIA

**“Recogida y análisis de los datos  
de un pulsioxímetro a través de  
una aplicación móvil”**

**TRABAJO FINAL DE GRADO**

Autor/a:

**Adrián Vaello Pérez**

Tutor/a:

**Jaime Lloret Mauri**

**GANDIA, 2022**

## Resumen y palabras clave

Un pulsioxímetro u oxímetro de dedo es un aparato médico que consigue monitorizar el nivel de concentración de oxígeno que tenemos en la sangre de manera no intrusiva. También indica la frecuencia cardíaca y el pulso del paciente. Actualmente es un dispositivo muy importante en hospitales y centros de salud porque ayuda a los médicos a saber, sin hacer pruebas muy exhaustivas, en qué estado se encuentra el paciente.

El presente proyecto consiste en el diseño y desarrollo de una aplicación móvil “bajo demanda” a la cual se le conectará un pulsioxímetro comercial de bajo coste a través de Bluetooth. Esta aplicación recogerá los datos enviados por el pulsioxímetro para posteriormente realizar su análisis y mostrarlos en la aplicación de manera fácil y visual. Los datos enviados por el pulsioxímetro nos permitirán calcular otros parámetros de gran utilidad clínica para poder detectar algunas de las anomalías cardiovasculares más comunes en los pacientes, como por ejemplo el índice de variabilidad pletismográfica, muy útil para saber la respuesta a la administración de fluidos que tendrá el paciente, la relación entre la curva pletismográfica y el pulso cardíaco, calculando amplitudes, áreas bajo la curva, etc, y calculando la cisura dicrótica, útil para saber la posible presencia de anemia.

Aparte de eso, la aplicación dispondrá de una funcionalidad que permitirá la descarga de los datos recogidos en formato Excel para su posterior análisis y estudio de estos por un médico especializado. Para esto se dispondrá de una pantalla de gestión de los pacientes desde la cual acceder a estos datos en formato Excel.

**Palabras clave:** Pulsioxímetro, bluetooth, datos, análisis, App, aplicación.

## Abstract and keywords

A pulse oximeter or finger oximeter is a medical device that manages to monitor the level of oxygen concentration that we have in the blood in a non-intrusive way. It also indicates the patient's heart rate and pulse. It is currently a very important device in hospitals and health centers because it helps doctors to know, without carrying out very exhaustive tests, what state the patient is in.

This project consists of the design and development of a mobile application "on demand" to which a low-cost commercial pulse oximeter will be connected via Bluetooth. This application will collect the data sent by the pulse oximeter to later perform its analysis and display it in the application in an easy and visual way. The data sent by the pulse oximeter will allow us to calculate other very useful parameters to be able to detect some of the most common cardiovascular anomalies in patients, such as the plethysmographic variability index, very useful to know the response to the administration of fluids that the patient will have, the relationship between the plethysmographic curve and the heart rate, calculating amplitudes, areas under the curve, etc., and calculating the dicrotic fissure, useful for determining the possible presence of anemia.

Apart from that, the application will have a functionality that will allow the download of the data collected in Excel format for later analysis and study of these by a specialized doctor. For this, a patient management screen will be available from which to access this data in Excel format.

**Keywords:** Pulse oximeter, bluetooth, data, analysis, App, application.

## Resum i paraules clau

Un pulsioxímetre o oxímetre de dit és un dispositiu mèdic que aconsegueix monitoritzar el nivell de concentració d'oxigen que tenim en sang d'una manera no intrusiva. També indica la freqüència cardíaca i el pols del pacient. Actualment és un dispositiu molt important als hospitals i centres de salut perquè ajuda els metges a saber, sense fer proves exhaustives, en quin estat es troba el pacient.

El present projecte consisteix en el disseny i desenvolupament d'una aplicació mòbil "sota demanda" sobre la qual es connectarà un pulsioxímetre comercial de baix cost a través de Bluetooth. Aquesta aplicació arrebregarà les dades enviades pel pulsioxímetre per a, posteriorment, realitzar la seua anàlisi i mostrar-los en l'aplicació de forma fàcil i visual. Les dades enviades pel pulsioxímetre ens permetran calcular altres paràmetres de gran utilitat clínica per a poder detectar algunes de les anomalies cardiovasculars més comunes en els pacients com, per exemple, l'índex de variabilitat pletismogràfica, molt útil per a saber la resposta a l'administració de fluïts que tindrà el pacient, la relació entre la curva pletismogràfica i el pols cardíac, tot calculant amplituds, àrees baix la curva, etc., i calculant la cisura dicròtica, molt útil per a saber la possible presència d'anèmia.

D'altra banda, l'aplicació disposarà d'una funcionalitat que permetrà la descàrrega de les dades arrebegades en format Excel per a la seua posterior anàlisi i estudi d'aquestes per part d'un metge especialitzat. Per a açò, es disposarà d'una pantalla de gestió dels pacients des de la qual poder accedir a aquestes dades en format Excel.

**Paraules clau:** Pulsioxímetre, bluetooth, dades, anàlisi, App, aplicació

# ÍNDICE

Resumen y palabras clave .....	2
Abstract and keywords .....	3
Resum i paraules clau.....	4
Índice de imágenes.....	6
Agradecimientos .....	7
Capítulo 1. Introducción .....	8
1.1 Presentación.....	8
1.2 Objetivos del proyecto .....	9
1.3 Estructura de la memoria.....	9
1.4 Precedentes del proyecto .....	10
1.4.1 Estado actual del tema del proyecto .....	11
Capítulo 2. Estado del arte.....	13
2.1 Estado del arte .....	13
Capítulo 3. Propuesta .....	15
3.1 Mockup .....	15
Capítulo 4. Desarrollo .....	17
4.1 Introducción al desarrollo .....	17
4.1.1 Control de versiones .....	17
4.2 Ingeniería inversa.....	18
4.3 Desarrollo general.....	22
4.3.1 Metodología de trabajo y herramientas de desarrollo.....	22
4.3.2 Estructura de la aplicación .....	23
4.3.3 Software y librerías .....	24
4.4 Problemas de implementación .....	25
Capítulo 5. Testeo .....	32
5.1 Testeo conmigo mismo .....	32
5.2 Testeo con personas ajenas.....	35
Capítulo 6. Manuales.....	37
6.1 Manual de instalación.....	37
6.2 Guía de uso.....	37
Capítulo 7. Evaluación.....	43
7.1 Cumplimiento del objetivo .....	43
7.2 Conclusiones sobre el proyecto.....	43
7.3 Aportaciones personales .....	44
7.4 Futuras líneas de trabajo .....	45
Capítulo 8. Referencias .....	46

## Índice de imágenes

En la Figura 1 se muestra el producto de Masimo rainbow SET .....	13
En la Figura 2 se puede ver el diagrama de flujo de la aplicación en versión móvil. ...	16
En la Figura 3 se puede observar la Gráfica de uso de GitHub.....	17
En la Figura 4 se puede observar la primera versión de la aplicación 1 .....	18
En la Figura 5 se muestra el oxímetro que se ha utilizado .....	19
En la Figura 6 se pueden ver los bytes recibidos en formato hexadecimal .....	20
En la Figura 7 se pueden ver los valores recibidos tras la conversión .....	21
En la Figura 8 se puede ver la estructura de directorios de la aplicación .....	23
En la Figura 9 se puede ver la estructura de la carpeta “res” .....	23
En la Figura 10 se puede ver un ejemplo del tipo de gráfica usado .....	25
En la Figura 11 se puede ver la fórmula de cálculo del PVi.....	27
En la Figura 12 se observa un ejemplo de cómo calcular Pimax y Pimin .....	27
En la Figura 13 se puede ver una imagen explicativa del cálculo del área bajo la curva. .....	29
En la Figura 14 se muestra una onda donde se muestra la cisura dicrótica .....	30
En la Figura 15 se muestra los diferentes tipos de vasoconstricción o vasodilatación según el valor del pico de la cisura dicrótica .....	31
En la Figura 16 se puede ver la aplicación en funcionamiento haciendo testeo. ....	32
En la Figura 17 se puede ver el Excel con los datos clínicos resultado de la prueba de testeo realizada. ....	33
En la figura 18 se puede ver el Excel con los datos recogidos de la gráfica.....	33
En la Figura 19 se ve la Splash Screen de la aplicación .....	37
En la Figura 20 se puede ver la página principal de la App.....	38
En la Figura 21 se puede ver la página del historial de pacientes de la App .....	39
En la Figura 22 se puede ver la página de consejos de la App .....	40
En la Figura 23 se puede ver la pantalla donde se muestran los datos.....	41
En la Figura 24 se puede ver la pantalla para descargar los datos del paciente. ....	42

## Agradecimientos

A mis padres y a mi tío.

# Capítulo 1. Introducción

## 1.1 Presentación

Un pulsioxímetro u oxímetro de dedo es un aparato médico que consigue monitorizar el nivel de concentración de oxígeno que tenemos en la sangre de manera no intrusiva. También indica la frecuencia cardíaca y el pulso del paciente. Actualmente es un dispositivo muy importante en hospitales y centros de salud porque ayuda a los médicos a saber, sin hacer pruebas muy exhaustivas, en qué estado se encuentra el paciente.

Desde el inicio de la pandemia COVID-19 hasta la actualidad ha aumentado mucho su nivel de uso debido a que ha resultado muy útil su utilización para detectar posibles casos positivos de COVID siendo las personas asintomáticas y muy útil también para el tratamiento de pacientes que deriven a plantas Ucis.

Por lo general, un nivel bajo de oxígeno en sangre provoca síntomas como fatiga o falta de aliento, síntomas muy comunes cuando se tiene COVID, por tanto, un pulsioxímetro de uso doméstico puede ayudar a saber si necesita atención médica incluso si no tiene síntomas, ya que puede ser que tenga un nivel bajo de oxígeno en sangre, pero no presentar ningún síntoma de dificultad respiratoria. Esto permitiría detectar tempranamente la denominada hipoxemia “silenciosa” a fin de tratar oportunamente al paciente y evitar complicaciones. Se postula que la monitorización domiciliaria con pulsioxímetros reduciría significativamente las admisiones a cuidados intensivos, las intubaciones y la tasa de mortalidad a causa de la COVID-19. [1]

El presente proyecto consiste en el diseño y desarrollo de una aplicación móvil “bajo demanda” a la cual se le conectará un pulsioxímetro comercial de bajo coste a través de Bluetooth. Esta aplicación recogerá los datos enviados por el pulsioxímetro para posteriormente realizar su análisis y mostrarlos en la aplicación de manera fácil y visual. Los datos enviados por el pulsioxímetro nos permitirá calcular otros parámetros de gran utilidad para poder detectar algunas de las anomalías cardiovasculares más comunes en los pacientes, como por ejemplo el índice de variabilidad pletismográfica, muy útil para saber la respuesta a la administración de fluidos que tendrá el paciente, la relación entre la curva pletismográfica y el pulso cardíaco, calculando amplitudes, áreas bajo la curva etc, y calculando la cisura dicrótica, útil para saber la posible presencia de anemia.

A parte de eso, la aplicación dispondrá de una funcionalidad que permitirá la descarga de los datos recogidos en formato Excel para su posterior análisis y estudio de estos por un médico especializado. Para esto se dispondrá de una pantalla de gestión de los pacientes desde la cual acceder a estos datos en formato Excel. Esta última permitirá tener un registro de los pacientes con todas las fechas en las que se le ha realizado una medición junto a los datos de estas.

Todo lo descrito anteriormente permitirá al enfermero/a o médico/a tener un primer análisis del paciente de manera muy rápida y efectiva. En apenas unos 7 segundos el enfermero/a podrá ya estar visualizando los datos y en base de estos poder obtener un veredicto suficientemente efectivo como para decidir enviar al paciente a realizar un estudio más intensivo o no.

Este proyecto está enmarcado dentro de un proyecto de investigación con la colaboración de la Universitat Politècnica de València (UPV) y la unidad de Servicio de Neumología del Hospital Arnau de Vilanova, Valencia, en concreto con el doctor José Belda Ramírez. La participación por parte de los colaboradores del hospital ha sido clave en el asesoramiento médico.

El presente documento describe todo el proceso que se ha seguido durante todo el desarrollo de la aplicación. Desde su inicio como proyecto hasta su validación final.

## 1.2 Objetivos del proyecto

Este proyecto se divide en varios objetivos principales y secundarios:

- Objetivo principal: Desarrollar una aplicación móvil que se conecte vía bluetooth con un pulsioxímetro de bajo coste.
- Objetivos secundarios:
  1. Crear una interfaz amigable e intuitiva.
  2. Recoger, analizar y mostrar los datos recibidos por el pulsioxímetro en la aplicación.
  3. Implementar en la aplicación los cálculos clínicos solicitados por el médico.
  4. Mostrar las gráficas de los datos obtenidos.
  5. Ahorrar en costes y tiempo de medición.
  6. Aplicación utilizable en móviles y tablets tanto en vertical como en horizontal.
  7. Implementar la posible descarga de los datos en formato Excel para su futuro estudio.

## 1.3 Estructura de la memoria

La memoria estará dividida en capítulos, cada uno explicará las diferentes partes del proyecto.

### **Capítulo 1: Introducción**

En el primer capítulo se realiza una breve introducción al proyecto presentando de que va a consistir este, se enumeran los objetivos principales del proyecto y se explicara si existe relación con otros proyectos, estado en el que se encuentra el tema a desarrollar, problemas existentes...

### **Capítulo 2: Estado del arte y entorno**

En el segundo capítulo se hará una investigación sobre los productos ya existentes similares al nuestro que haya en el mercado, así como adentrarnos un poco en cómo funcionan estos.

### **Capítulo 3: Propuesta**

En el tercer capítulo se explicará en qué consistirá el proyecto, así como las partes que tendrá. También se explicará la propuesta de aplicación inicial que me hicieron por parte de la persona interesada en el proyecto, qué aspectos propuse y de mejora y cómo lo llevé a cabo.

### **Capítulo 4: Desarrollo**

En el cuarto capítulo primeramente se hará una pequeña introducción a como se empezó el desarrollo del proyecto, así como explicar cómo se han controlado las diferentes versiones del código y de la aplicación. Posteriormente se explicarán los diferentes pasos que he seguido para la elaboración del proyecto, desde la ingeniería inversa del código y pulsioxímetro, pasando por los pasos de desarrollo y qué herramientas he usado, hasta la estructura general de la aplicación y qué librerías/software he usado.

Finalmente, expondré algunos de los problemas que he tenido y como he conseguido solucionarlo o no.

### **Capítulo 5: Testeo**

En el quinto capítulo desarrollaré los pasos que he seguido para el testeo de la aplicación, tanto por la parte médica como por la visual e interactiva. Para ello contaré con diferentes personas de diferentes estudios y conocimientos médicos y tecnológicos, simulando así su uso tanto médico como casero, para que prueben su uso y así recoger datos sobre si su uso es fácil o no.

### **Capítulo 6: Manuales**

En el sexto capítulo se explicará dónde encontrar la aplicación, cómo instalarla y cómo usarla de forma general.

### **Capítulo 7: Conclusiones**

En este capítulo, se realizará una valoración final del proyecto y el aprendizaje individual.

### **Capítulo 8: Referencias**

En este último capítulo se listarán todas las referencias consultadas, así como páginas o enlaces importantes para el entendimiento del trabajo.

## **1.4 Precedentes del proyecto**

Para el estudio de los precedentes similares a este proyecto se ha accedido a la web de "riunet" [2] en busca de otros trabajos académicos que tengan similitudes.

## 1.4.1 Estado actual del tema del proyecto

Tras investigar los diferentes trabajos académicos publicados en “riunet” he encontrado algunos trabajos que sirven de precedente a nuestro proyecto.

### **Diseño e implementación de un pulsioxímetro [3]**

Es un trabajo de fin de grado hecho por David López Serrano presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València en el año académico 2016-2017.

Este trabajo se centra en el desarrollo tanto de la parte software como el diseño hardware de un pulsioxímetro, pero también se diseñó una pequeña aplicación en Android para que los pacientes puedan ver sus mediciones.

Con este proyecto se consiguió, gracias a su programación y diseño del hardware, disminuir el consumo del pulsioxímetro al mínimo, apagando los LED's cuando no se están usando. También se consigue un ahorro económico en la parte del hardware al no tener que almacenar los datos en el dispositivo.

Se cumplieron todos los objetivos generales que se describieron al comienzo del proyecto, por tanto, se puede dar por finalizado con éxito.

Es un proyecto similar al nuestro pero cuyo trabajo principal está centrado más en la parte de diseño y programación del pulsioxímetro y no tanto en la programación de la aplicación, como es en nuestro caso.

### **Desarrollo de un sistema IOT integrado con dispositivos eHealth para la detección automática de la variabilidad cardíaca [4]**

Es un Trabajo Fin de Grado hecho por Carmen Palao Cruz, presentado en la Escuela Técnica Superior de Ingenieros de Telecomunicación de la Universitat Politècnica de València en el curso académico 2016-2017.

El trabajo consiste en el diseño e implementación de un sistema de comunicaciones que permita a los pacientes con problemas cardiovasculares llevar a cabo una monitorización constante de su ritmo cardíaco.

En este proyecto, al igual que el anterior, se centra en el diseño e implementación del pulsioxímetro y no tanto en la aplicación. En este caso también se diseñó una aplicación para poder mostrar algunos datos y la gráfica del ritmo cardíaco, pero se trata de una aplicación simple y que para que esta reciba los datos necesita estar conectada físicamente, mediante un cable USB, al pulsioxímetro.

Tras esta investigación de proyectos anteriores se puede ver que el tema de este proyecto apenas ha sido investigado, y cuando se ha hecho se le ha dado más

importancia al diseño y programación del pulsioxímetro que en ofrecer una aplicación realmente útil tanto a los profesionales médicos como a los usuarios normales. En el repositorio de "riunet" estos son los dos únicos trabajos académicos que tienen una cierta relación con el nuestro. Por tanto, el estado actual del proyecto y del tema a tratar aún está en investigación ya que no se ha conseguido aún tener una aplicación que cumpla con los requisitos y objetivos de la nuestra.

## Capítulo 2. Estado del arte

Hoy en día, debido a los numerosos recortes en sanidad que se han ido produciendo a lo largo de los años, cada vez es más difícil comprar el material necesario para poder atender correctamente a todos los pacientes como se merecen. En la actual pandemia del COVID-19 todos hemos podido comprobar a lo que puede llevar la falta de escasez de material en hospitales y Ucis, quién sabe las vidas que se hubieran podido salvar si este proyecto hubiera salido antes. Numerosas fuentes afirman que el tener un pulsioxímetro en casa puede salvar muchas vidas. [1]

Por eso nuestro proyecto ofrece una aplicación que, conectada a un pulsioxímetro super económico, nos permite obtener los parámetros clínicos más importantes a la hora de hacer un primer análisis a un paciente.

### 2.1 Estado del arte

Para encontrar precedentes de este proyecto se ha realizado una búsqueda intensiva en internet para encontrar proyectos similares. Se han encontrado dos proyectos similares al que se pretende realizar con este.

#### **Masimo rainbow SET [5]**

Masimo rainbow SET es una plataforma de monitorización no invasiva que incluye la oximetría de pulso con medición en condiciones de movimiento y baja perfusión. Máximo rainbow SET permite medir múltiples parámetros como son Saturación de oxígeno (SpO<sub>2</sub>), frecuencia cardíaca (PR), índice de perfusión (Pi), índice de variabilidad pletismográfica (PVi), Frecuencia respiratoria (RR), entre otros.



En la Figura 1 se muestra el producto de Masimo rainbow SET

Masimo rainbow SET es un proyecto cuya finalidad es similar a la nuestra pero que usa dispositivos mucho más caros y cuyo producto de venta es el propio medidor de los parámetros.

### **TFG aplicación para el parseo y la visualización de los datos de un pulsioxímetro [6]**

Este proyecto consiste en un trabajo fin de grado realizado por Mikel Aingeru en el curso 2017-2018 en la universidad del País Vasco. Consiste en el desarrollo de una pulsera que sea capaz de recoger datos para el posterior muestreo de la señal pletismográfica y su posterior envío vía Bluetooth. También se desarrolló una aplicación móvil para visualizar los datos recibidos por la pulsera y mostrar una gráfica dinámica en tiempo real.

El registro de versiones se guardó en un repositorio de GIT. [7]

Finalmente, en este TFG no se consiguió cumplir los objetivos que se propusieron ya que no se consiguió hacer la conexión entre la pulsera y la aplicación vía Bluetooth por problemas en la pulsera.

Como hemos visto en los dos proyectos anteriores, es un tema en el cual el desarrollo aún está poco avanzado. Prácticamente los proyectos que han salido adelante con un tema parecido al mío se pueden contar con los dedos de una mano y la mayoría de ellos han tenido problemas en alguna parte del desarrollo, y los que no han tenido problemas, son proyectos que utilizan una tecnología fuera de nuestro alcance y objetivo. Por tanto, nuestro proyecto creo que puede tener un hueco en el mercado.

## Capítulo 3. Propuesta

Este TFG se engloba dentro del proyecto polisabio 2020/A43 con el título “Estimación de nivel de riesgo cardiovascular en obesidad infantil mediante un modelo estadístico” enviado en “FISABIO-UPV” en la convocatoria de ayudas POLISABIO 2019 y como entidad financiadora la Fundación para el Fomento de la Investigación Sanitaria y Biomédica de la Comunitat Valenciana. [8] Consiste en una propuesta que se nos hizo por parte del médico José Belda Ramírez del Hospital Arnau de Vilanova en el Servicio de Neumología de Valencia, quien pensaba que se podría obtener los mismos parámetros de medida de un pulsioxímetro más barato conectado a una aplicación que realizara algunos cálculos extra que de una máquina convencional que son más grandes, engorrosas y caras. Por tanto, se hizo esta propuesta y se empezó el desarrollo.

Los objetivos principales que se marcaron era el poder leer los datos de un pulsioxímetro barato en una aplicación y que esta fuera capaz de, gracias a los datos ofrecidos por el pulsioxímetro, poder calcular otros parámetros de gran relevancia clínica como puede ser el PVi (índice de variabilidad pletismográfica) o la cistura dicrótica. Aparte también se deberían poder descargar estos datos recogidos en un formato Excel por si se necesitaran para un estudio más intensivo.

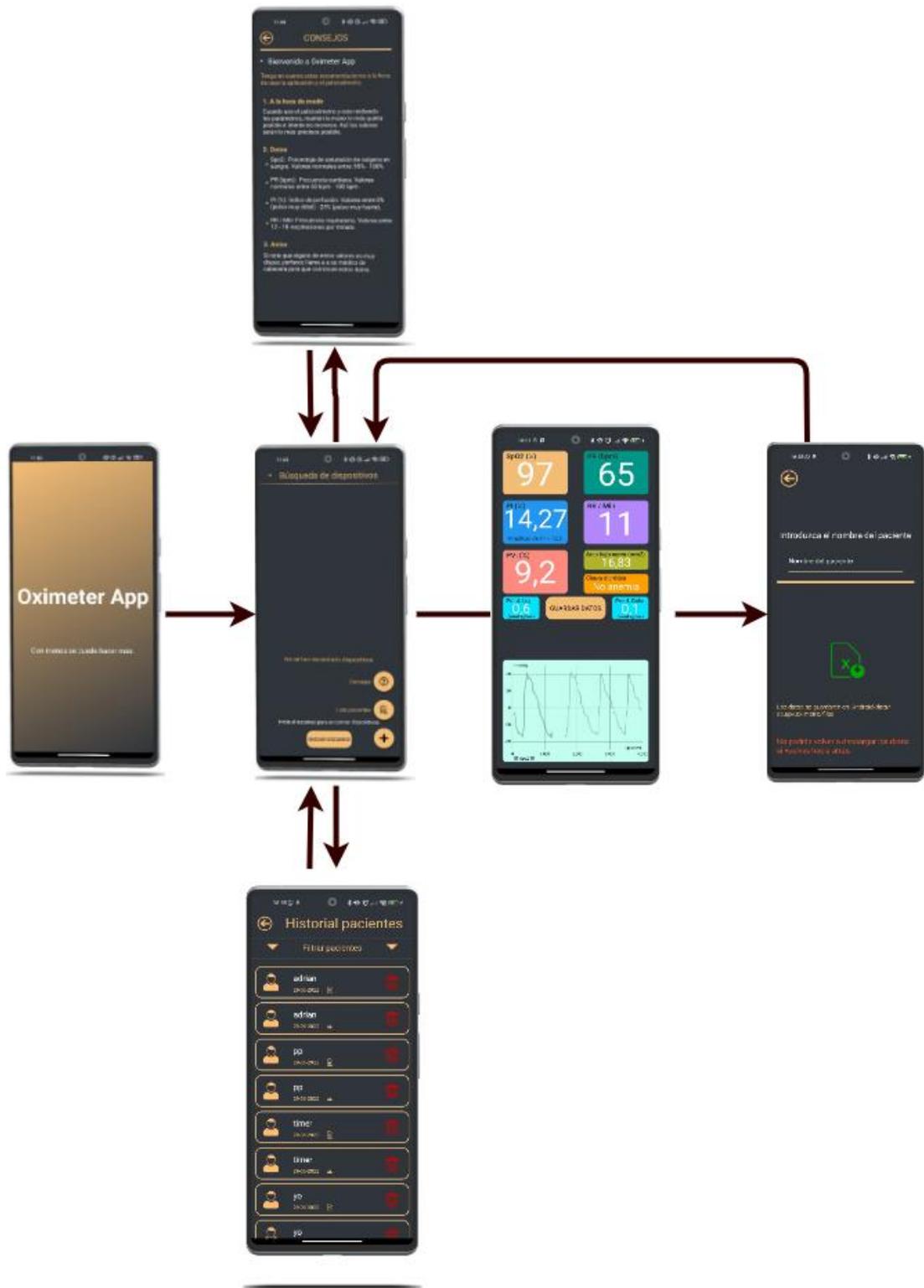
Para el desarrollo de la aplicación se escogió el entorno de programación Android Studio. Actualmente es el entorno más usado para el desarrollo de aplicaciones móviles para Android. Como lenguajes de programación principales se usarán Java Android para las interacciones de la página, realizar y mostrar los parámetros necesarios y XML como lenguaje para el diseño de las páginas.

### 3.1 Mockup

Tras las diferentes reuniones con el cliente y el análisis de los objetivos, se empezó a trabajar en el diseño de la aplicación, es decir, cómo debería lucir, qué colores escoger, cómo se deben mostrar los datos en pantalla, etc.

Se pensó que la aplicación no solo debería ser visible para móviles, sino que también se debería poder visualizar correctamente en dispositivos de un tamaño más grande, como, por ejemplo, tablets. Para ello el diseño se debería ajustar a estos tamaños tanto en formato vertical como en horizontal.

La aplicación debía ser sencilla, que en pocos pasos puedas ya estar usándola a pleno rendimiento, la visualización de los datos debía ser sencilla y concisa, sin elementos irrelevantes, y tanto la descarga de los datos de los pacientes como la visualización debían ser sencillas y entendibles por todos los públicos. Por tanto, se desarrolló un diagrama de navegación de las diferentes páginas que finalmente compondrían nuestra aplicación.



En la Figura 2 se puede ver el diagrama de flujo de la aplicación en versión móvil.

# Capítulo 4. Desarrollo

## 4.1 Introducción al desarrollo

Este proyecto se propuso y se empezó a desarrollar en el 2019 por parte de otras personas, pero por causas que desconozco se paró su desarrollo hasta que al inicio del 2022 se me propuso continuarlo, ya que las personas que lo encomendaron seguían interesadas en continuarlo y acabarlo.

### 4.1.1 Control de versiones

Durante el desarrollo de este proyecto se han usado diversas aplicaciones para el control del código y del desarrollo en general. En este caso he usado GitHub para el control de las diferentes versiones del código, usando GitHub Desktop para el manejo de las funciones que ofrece GitHub. [9]

GitHub es una forja para alojar proyectos utilizando el sistema de control de versiones Git. Se utiliza principalmente para la creación de código fuente de programas de ordenador. Desde el comienzo de la pandemia COVID-19 GitHub ha casi triplicado su número de repositorios que se han creado, cosa que lo convierte en el sistema de repositorios de código que más se usa en el mundo.



En la Figura 3 se puede observar la Gráfica de uso de GitHub

En mi caso el GitHub se organiza en 3 repositorios claramente diferenciados:

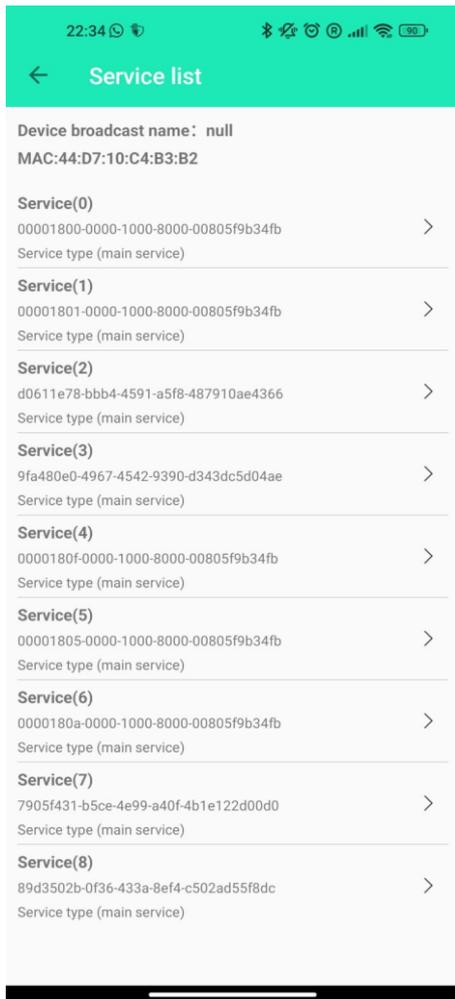
- Main: Es donde se aloja el código principal y donde se alojará el código final de la aplicación.
- Pruebas: En este repositorio es donde se hacen más cambios. Aquí se van alojando todos los cambios pequeños que se van haciendo en la aplicación.
- Versión 0.X: Cuando la aplicación ya tiene una serie de cambios, guardados en pruebas, entonces se hace un guardado en esta rama para así tener una versión de código funcional con todos los cambios ya incluidos. Se guardan usando el formato versión 0.X donde X es el numero de la versión.

## 4.2 Ingeniería inversa

Cuando empecé el proyecto se me dio una mínima base de la aplicación ya hecha. La aplicación lucía así:



En la Figura 4 se puede observar la primera versión de la aplicación 1



En la Figura 5 se puede observar la primera versión de la aplicación 2

Por tanto, mi primer trabajo era hacer un poco de ingeniería inversa del código dado y del pulsioxímetro que me proporcionaron para ver hasta qué estado de desarrollo se había llegado y como se estaba desarrollando.

El modelo de oxímetro elegido era el siguiente [10]:



En la Figura 5 se muestra el oxímetro que se ha utilizado

Antes que nada, lo que debía descubrir era cómo se enviaban los datos desde el sensor, por tanto, lo primero que hice fue descargarme una aplicación de escáner BLE. En mi caso fue BLE scanner, que se puede encontrar en Play store y Apple store fácilmente. [11]

Al conectar el pulsioxímetro me di cuenta de que enviaba paquetes de 6 bytes muy frecuentemente, pero cada segundo aproximadamente enviaba un paquete de 12 bytes donde parecía que ahí se enviaban los datos interesantes para el estudio.

El siguiente paso ahora era encontrar un software de escáner BLE que se pudiera introducir en la aplicación. Primeramente, se probó con el software que nos ofrece Google. [12]

Pero, por razones que desconozco, fue imposible conectarlo con el oxímetro seleccionado, por tanto, tocaba buscar un sustituto a este.

Buscando por internet encontré un escáner llamado FastBLE. [13]

Este software sí que nos funcionó y por tanto ya podíamos recibir datos del pulsioxímetro. Antes comentaba que la aplicación que me ofrecieron estaba ya un poco empezada. Pues tras comparar la aplicación suministrada con la que nos ofrece FastBLE descubrí que la aplicación que me ofrecieron apenas tenía diferencias con la de FastBLE, es decir, solo estaba implementado el código de ejemplo de FastBLE. [14]

El siguiente paso era ver los datos que se enviaban del pulsioxímetro al móvil.

Tras conseguir captar los bytes recibidos conseguimos empezar a ver algo.

```
Valores :1653150874188; fa 06 80 00 00 86
Valores :1653150874230; fa 06 80 00 00 86
Valores :1653150874235; fa 06 80 00 00 86
Valores :1653150874267; fa 06 80 00 00 86
Valores :1653150874270; fa 06 80 00 00 86
Valores :1653150874314; fa 06 80 00 00 86
Valores :1653150874361; fa 06 80 00 00 86 fa 06 80 00 00 86
Valores :1653150874394; fa 06 80 00 00 86
Valores :1653150874427; fa 06 80 00 00 86
Valores :1653150874432; fa 06 80 00 00 86
Valores :1653150874462; fa 06 80 00 00 86
Valores :1653150874465; fa 06 80 00 00 86
Valores :1653150874497; fa 06 80 00 00 86
Valores :1653150874540; fa 06 80 00 00 86 fa 06 80 00 00 86
Valores :1653150874564; fa 06 80 00 00 86
Valores :1653150874595; fa 06 80 00 00 86
Valores :1653150874600; fa 06 80 00 00 86
Valores :1653150874628; fa 06 80 00 00 86
Valores :1653150874663; fa 06 80 00 00 86
Valores :1653150874696; fa 06 80 00 00 86
Valores :1653150874699; fa 06 80 00 00 86
Valores :1653150874727; fa 06 80 00 00 86
Valores :1653150874730; fa 06 80 00 00 86
Valores :1653150874764; fa 06 80 00 00 86 fa 06 80 00 00 86
Valores :1653150874795; fa 06 80 00 00 86
```

*En la Figura 6 se pueden ver los bytes recibidos en formato hexadecimal*

Pero claro, los datos así tal cual como llegaban no eran legibles ni se podían interpretar. Por tanto, había que encontrar la forma de conseguir esos datos, pero de una forma legible. Tras conseguir pasar los datos de formato hexadecimal a byte ya empezamos a ver algo más legible.

```
Valores: [-6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 11, -127, 0, 0, 0, 0, 0, 0, 1, -115]
Valores: [-6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122, -6, 6, -128, 0, 0, -122]
Valores: [-6, 6, -128, 0, 0, -122]
```

*En la Figura 7 se pueden ver los valores recibidos tras la conversión*

Al conseguir estos datos ya era hora de empezar a pensar en el desarrollo de la aplicación, que elementos debía tener, etc.

## 4.3 Desarrollo general

Como se ha comentado en el anterior apartado la aplicación estaba casi con solo el ejemplo proporcionado por FastBLE [14] , por tanto, me tocaba completar la aplicación casi desde cero.

### 4.3.1 Metodología de trabajo y herramientas de desarrollo

Para la organización del trabajo en el proyecto se ha usado la metodología “scrum”.

Scrum es un método de trabajo muy utilizado en el mundo del desarrollo de software. Es un método en el que se aplican de manera regular un conjunto de buenas prácticas con el fin de obtener el mejor resultado posible de los proyectos. Se caracteriza por:

- Una estrategia de desarrollo incremental en lugar de la planificación y ejecución completa del producto. Desarrollo mediante “sprints”.
- Basar la calidad del resultado más que en la calidad de los procesos empleados.
- Solapar diferentes partes del desarrollo.

En mi caso al trabajar yo solo en este proyecto me he organizado de tal forma que la cantidad de trabajo se ajuste al tiempo disponible que tenga en ese momento para dedicarle al proyecto. Los “sprints” los tenía más o menos cada tres semanas en donde me marcaba unos objetivos que quería tener cumplidos al acabar esas tres semanas.

Antes de empezar el proyecto se hizo una reunión con el tutor y una enfermera encargada de hacer de intermediaria entre el interesado y nosotros para definir los requisitos principales de la aplicación, los parámetros nuevos que había que mostrar o calcular y se estableció que el guardado de los datos debía ser en formato Excel para que posteriormente estos pudieran ser descargados y analizados. El diseño e interacción fue todo libertad mía de elección, confiaron en que lo sabría hacer bien. Lo único que me dijeron es que fuera una aplicación fácil de usar porque no todos los técnicos médicos tienen un nivel de tecnología medio o alto, por tanto, la aplicación debía tener todos los extras que creyera necesarios, pero con facilidad de uso.

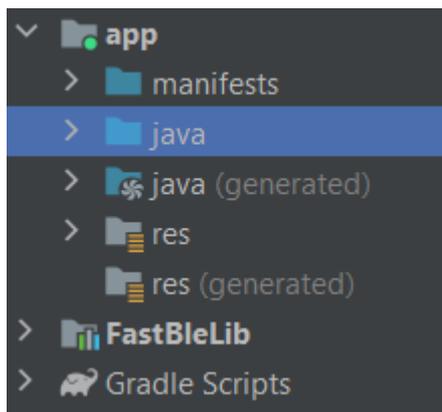
Para el desarrollo de la aplicación se han usado varias herramientas de desarrollo como son:

- Android Studio: Es el software que más se ha usado. Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android. Es el entorno donde toda la aplicación se ha desarrollado y probado. El lenguaje de programación usado es Java Android para la funcionalidad y XML para el diseño de las interfaces.

- Photoshop: Es el software de edición de imágenes más conocido y usado en el mundo. En el proyecto se ha usado para la creación o retoque de los iconos.
- PngTree y PngWing: Son páginas de internet que ofrecen iconos en formato .png gratuitos. La mayoría de los iconos de la aplicación se han cogido de estas páginas.

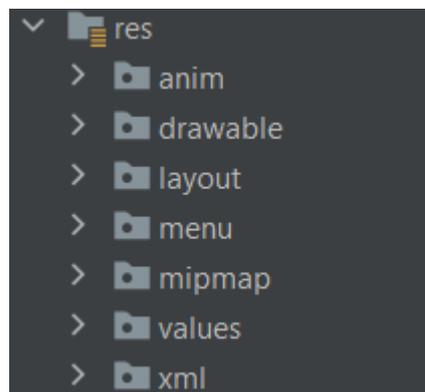
### 4.3.2 Estructura de la aplicación

La aplicación principalmente se divide en 5 directorios:



En la Figura 8 se puede ver la estructura de directorios de la aplicación

- La carpeta “manifests” en donde se incluye el fichero AndroidManifest.xml es la parte de la aplicación donde se declaran los diversos permisos que necesita la aplicación para por ejemplo poder usar el bluetooth del dispositivo. También aquí se declaran todas las pantallas que tendrá la aplicación.
- En la carpeta “Java” es donde se aloja todo el código Java, imprescindible para el funcionamiento de la aplicación. Aquí está el código que recoge los datos del sensor, donde se muestran estos datos, el código de guardado de los datos en Excel etc.
- La carpeta “res” contiene todos los recursos sin código, como diseños XML de las páginas, strings de IU, strings de texto, colores e imágenes e iconos de la aplicación entre otros.



En la Figura 9 se puede ver la estructura de la carpeta “res”

- En la carpeta "Gradle Scripts" es donde se implementan todas las librerías que se usan en el proyecto, se establecen las versiones de código y aplicación, el SDK mínimo de la aplicación entre muchas más cosas.
- Por último, se encuentra un directorio llamado "FastBleLib" que se trata de la librería FastBle que se ha usado para la conexión bluetooth entre el pulsioxímetro y la aplicación. Allí se encuentran todas las funciones que nos ofrece este software, más un ejemplo de demostración.

### 4.3.3 Software y librerías

#### **FastBle** [13]

Como ya se ha mencionado anteriormente, para la conexión entre el pulsioxímetro y la aplicación se ha usado el software FastBle.

FastBle es un software creado por el usuario Jasonchenlijian que ha subido este proyecto a su repositorio público de GitHub que consiste en el uso del software proporcionado por Google "BLE" pero que este usuario lo ha modificado para mejorar ciertas partes que para él se quedaban escasas en el software de Google. Algunas de las funciones características son:

- Filtrar, escanear, leer, escribir, suscribir y cancelar de forma sencilla.
- Soporta la adquisición de la intensidad de señal y el poder establecer una unidad de transmisión máxima.
- Permite reglas de escaneo personalizadas.
- Admite conexiones múltiples de varios dispositivos.

También tiene como ventaja que, al ser un software creado por un usuario particular, cualquiera puede dejar los problemas que ha tenido o problemas que ha encontrado y el creador le puede ayudar. Incluso el código se va actualizando permanentemente para corregir errores o "bugs".

#### **XSSF** [15]

Apache POI es una API Java multiplataforma de código abierto y gratuita escrita en Java. Apache POI proporciona API para que los programas Java lean y escriban archivos en formato Microsoft Office (Excel, WORD, PowerPoint, Visio, etc.). POI es el acrónimo de "Implementación de ofuscación deficiente", que significa "implementación difusa concisa". Dentro de la estructura Apache POI hay unos cuantos paquetes como HSMF para Outlook, HWPf para Word, entre otros. En nuestro caso hemos usado el XSSF.

XSSF es un paquete de Apache POI que proporciona la función de leer y escribir archivos de formato Microsoft Excel OOXML XLXS, es decir, la última versión de Microsoft Excel.

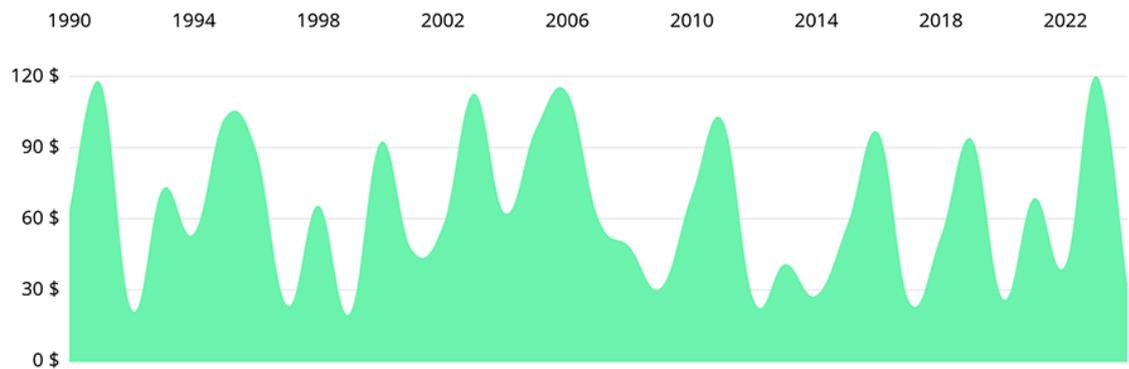
Los denominados Workbooks son las interfaces que se deben implementar que deben implementar para poder acceder a las funciones que nos ofrece XSSF.

En este proyecto se ha usado XSSF para escribir los documentos Excel que contienen los datos recogidos por el pulsioxímetro de un paciente y dividirlos en filas y columnas.

### **MPAndroidChart** [16]

MPAndroidChart es una librería de vista de gráficos de Android creada por PhilJay que admite gráficos de barras lineares, circulares, de radar, de burbujas, así como crear gráficos animados entre otras funcionalidades.

En este proyecto se ha usado para mostrar la gráfica del ritmo cardíaco. Dentro de la nomenclatura de la librería estaríamos usando un "LineChart", con la única diferencia de que nuestro "LineChart" es un gráfico animado, es decir, cuando la gráfica llega a su máximo en la parte derecha, los datos de la parte izquierda se empezaron a substituir por otros nuevos dando la sensación de movimiento.



*En la Figura 10 se puede ver un ejemplo del tipo de gráfica usado*

Esta librería al igual que FastBle al ser creada por un usuario normal, esta va recibiendo actualizaciones de forma más o menos constantes añadiendo nuevas funciones o simplemente corrigiendo errores.

## 4.4 Problemas de implementación

Durante el desarrollo de la aplicación me he encontrado con diversos problemas a la hora de programar algunas partes de la aplicación y que he tenido que ir solucionando poco a poco y buscando e informándome mucho. Aquí explicaré los mayores problemas y los que más tiempo me han llevado de solucionar.

### **Guardar y mostrar fichero desde la memoria interna**

Unos de los requerimientos que me dijeron en la primera reunión que tuvimos para hablar sobre el proyecto fue que los datos se tenían que poder guardar en formato Excel. Mi recomendación fue que estos se guardaran en un servidor en la nube para su mayor accesibilidad, pero estos me dijeron que eso supondría

mayor gasto y que por tanto el objetivo de que el proyecto sea lo más asequible posible se truncaba, entonces les propuse la posibilidad de guardar los datos en el propio dispositivo, propuesta que aceptaron.

Guardar los datos en local en el móvil no fue realmente un reto ya que Java ya nos proporciona una funcionalidad para hacer esto, el problema es dónde lo guarda.

Por defecto la carpeta donde se suelen guardar los datos en Android es: */storage/emulated/0/Android/data/es.upv.oximetro/files*.

Donde */storage/emulated/0/Android/data* es la dirección donde Android guarda los datos de la mayoría de aplicaciones que tenemos descargadas en el teléfono y *es.upv.oximetro/files* es la carpeta con el nombre del proyecto/la carpeta donde se guardarán los ficheros.

El problema que me encontré es que a partir de Android 10, Android deshabilitó el permiso por el cual tú podías acceder a esta carpeta y poder leer los datos que hay allí alojados. En versiones anteriores de Android esto sí que se podía acceder con unos simples permisos. Por tanto, tocaba buscar la forma de acceder a esta carpeta y obtener permisos de lectura.

Buscando en foros por internet encontré una librería interna de Android llamada "File provider". [17] Esta librería no hacía concretamente lo que yo buscaba que era el tener permiso de lectura en la carpeta mencionada anteriormente, pero en este momento me di cuenta de que realmente el tener acceso de lectura no era necesariamente lo que necesitaba era el poder abrir los ficheros con cualquier aplicación compatible con el formato Excel. Y eso es lo que me ofrecía file provider.

File provider ofrece, temporalmente, permisos de lectura y escritura en la dirección que se quiera. Para ello se tenían que dar una serie de permisos y cuando se intentara abrir ese fichero se debía de pasar la dirección de los ficheros en formato "Uri" usando este file provider junto con el tipo de ficheros que se quería abrir.

Finalmente, conseguí resolver este problema y ya se puede guardar los ficheros en la memoria del teléfono y acceder a estos cuando se desee con tan solo un "click".

### **Calculo Índice de Variabilidad Pletismográfica (PVI) [18]**

El cálculo de PVi era uno de los objetivos más importantes de este proyecto ya que era un valor de mucha importancia clínica que el pulsioxímetro no te ofrece.

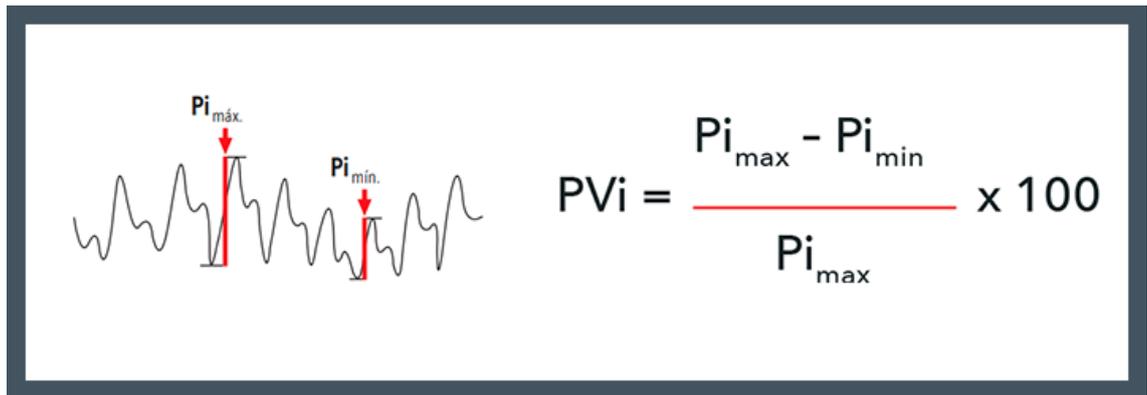
El índice de variabilidad pletismográfica es un parámetro no invasivo que ayuda a los profesionales médicos a monitorizar la respuesta que un paciente puede tener a la administración de fluidos. El cálculo se basa en los cambios medidos en el Pi (índice de perfusión) durante un ciclo.

Un ajuste no adecuado del fluido puede provocar una hipovolemia o hipervolemia. Cuando mayor es el valor del PVi más posibilidades hay que el paciente responda a la administración de fluidos.

El PVI se calcula de la siguiente forma: [19]

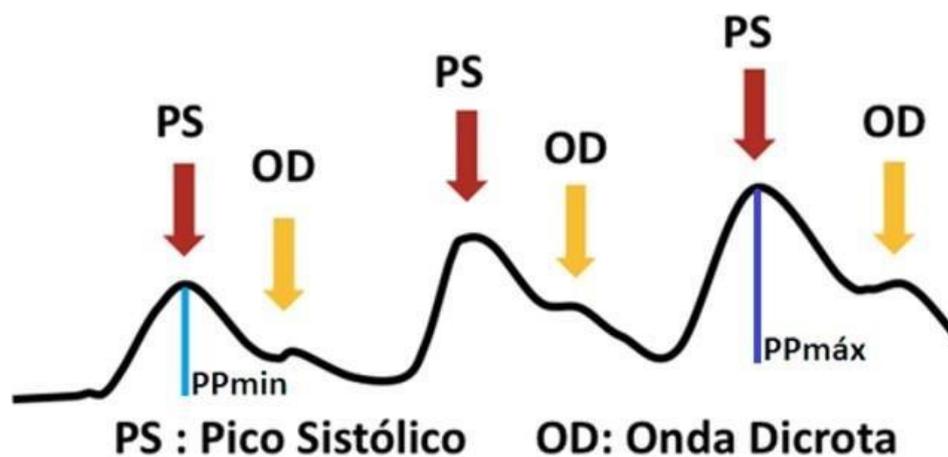
$$PVi = \frac{P_{imax} - P_{imin}}{P_{imax}} * 100$$

(1)



En la Figura 11 se puede ver la fórmula de cálculo del PVI

En mi caso el problema no era el hecho de aplicar la fórmula para calcular el PVI, sino que lo difícil y donde tuve problemas es en encontrar el  $P_{imax}$  y el  $P_{imin}$  a partir de los datos de la gráfica. Había que encontrar la amplitud máxima y mínima de un mismo ciclo solo teniendo en cuenta los picos sistólicos y descartando los máximos, mínimos y amplitudes de las ondas dicrotas.



En la Figura 12 se observa un ejemplo de cómo calcular  $P_{imax}$  y  $P_{imin}$

Tras meses probando diferentes métodos para el cálculo finalmente encontré uno que parecía ser el perfecto. Se trataba de ir calculando poco a poco los datos usando un método de ventana deslizante. Por ejemplo, si se usa una ventana de 10, primero se cogerán los valores del 0 al 9 y, primeramente, se mirará la tendencia inicial de la gráfica de forma muy fácil, si el valor de la posición 9 es

mayor que el de la posición 0 entonces está subiendo, si no, está bajando. Finalmente, se calculará la suma de todos los 10 valores de esa ventana para compararlo después.

Después pasaríamos a la ventana que incluye los valores en las posiciones 1 a 10 y ahora para calcular la tendencia de la recta se comparan la suma de los valores entre las posiciones 0 a 9 con la suma de los valores entre las posiciones 1 a 10. Si esta última es más grande quiere decir que la gráfica sigue subiendo y por tanto no hay que calcular nada, si fuera más pequeño, quiere decir que ahora está bajando la gráfica y que por tanto hemos pasado un máximo que se calculará mirando el máximo de los valores de las posiciones 1 a 10. Seguiríamos calculando de esta forma hasta que se acabe el ciclo.

De esta forma conseguimos librarnos de los máximos y mínimos de las ondas dicrotas, porque, aunque los valores dentro de la ventana sí que aumenten, la suma de los valores de la ventana siempre seguirá siendo menor a la suma de los valores de la ventana anterior, por tanto, seguirá la tendencia a la baja y no calculará máximos ni mínimos.

Hay algunas variables externas que no dependen del código pero que pueden afectar mucho al resultado de este valor como por ejemplo si el pulsioxímetro no está bien colocado en el dedo o si durante las mediciones de los valores esa persona se mueve o mueve la mano. Todos estos valores afectan a la gráfica haciendo que haga cosas poco normales.

También un problema relacionado con el cálculo del PVi es que los datos que nos envía el sensor para mostrar la gráfica están en complemento a dos.

El complemento a dos es una forma de almacenar números enteros, la cual se utiliza para representar números negativos en sistema binario.

Por tanto, tanto para mostrar la gráfica como para calcular el PVi había que pasar estos valores a valores decimales normales. [20]

### **Normalización de los datos de la gráfica**

Los datos que el pulsioxímetro nos ofrece para el muestreo de la gráfica están compuestos por valores positivos como negativos, cosa que no tiene mucho sentido porque es un poco extraño encontrar un valor de presión del oxígeno negativo. Por tanto, para el muestreo del gráfico se tuvieron que normalizar los datos para mostrarlos todos positivos manteniendo el sentido de la gráfica.

La estandarización o normalización de datos significa escalar los datos a un pequeño intervalo específico. En mi caso use la normalización por min-max.

El método de normalización min-max es transformar linealmente los datos originales. La fórmula es:

$$X_n = \frac{X - X_{min}}{X_{max} - X_{min}} * (N_{max} - N_{min}) + N_{min}$$

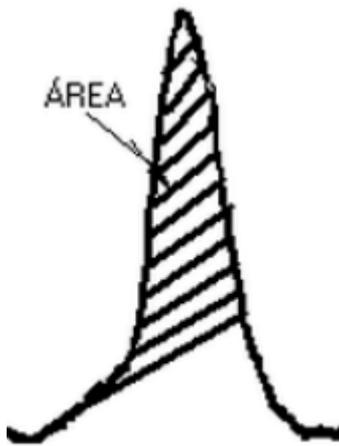
(2)

Siendo  $X_n$  el nuevo valor normalizado,  $X$  el valor sin normalizar,  $X_{min}$  y  $X_{max}$  el mínimo y máximo de los valores sin normalizar y  $N_{max}$  y  $N_{min}$  los nuevos máximo y mínimo a los que se quiere normalizar los valores.

### Cálculo del área bajo la curva

El cálculo del área bajo la curva ha sido uno de los problemas más difíciles que me he encontrado en el desarrollo, no por la dificultad de calcular el área, sino por saber de qué función calcularla o sobre qué curva calcularla.

El cálculo del área se debe hacer en cada ciclo respiratorio, es decir, en cada inhalación y exhalación o lo que es lo mismo, en cada subida y bajada de la gráfica. Para ello he usado una de las partes de la función para calcular el PVI. En esta función para calcular las amplitudes de cada ciclo respiratorio estas se hacen cuando la curva cambia de tendencia, es decir, cuando la gráfica llega a su punto más mínimo, entonces es ahí cuando aprovechamos para calcular el área con todos los puntos que forman la gráfica hasta ese punto.



*En la Figura 13 se puede ver una imagen explicativa del cálculo del área bajo la curva.*

El primer problema que me he encontré es que la gráfica no es regular, es decir, tanto la primera parte de la gráfica como la final no comparten mínimo, por tanto, el cálculo aún se nos hacía más difícil. Y el segundo problema es que no conocemos la función de la gráfica, y en caso de conseguirla, esta será diferente para cada paciente. Por tanto, había que buscar otra forma de calcular áreas sin saber la función y sin que compartan mínimo, pero que el resultado sea lo más ajustado posible.

Investigando por internet encontré una forma de calcular áreas usando rectángulos sobre los cuales se calculan sus áreas y sumando todas las áreas se puede aproximar bastante bien al área bajo la curva real. [21]

Para calcular el área de estos rectángulos primero tenemos que definir la base y la altura de estos. Como este valor se calculará al mismo tiempo que las amplitudes para el cálculo de PVI, en este momento tenemos también el valor de mínimo relativo en esta parte de la gráfica, por tanto, el valor de la altura será la diferencia entre cada valor de la gráfica menos el mínimo relativo.

$$H = \text{DatoGrafica} - \text{Min}$$

(3)

Para calcular la base la he usado usando el tiempo total del ciclo de muestras, en nuestro caso 4000 ms y el número total de valores en el ciclo completo. La fórmula en este sería:

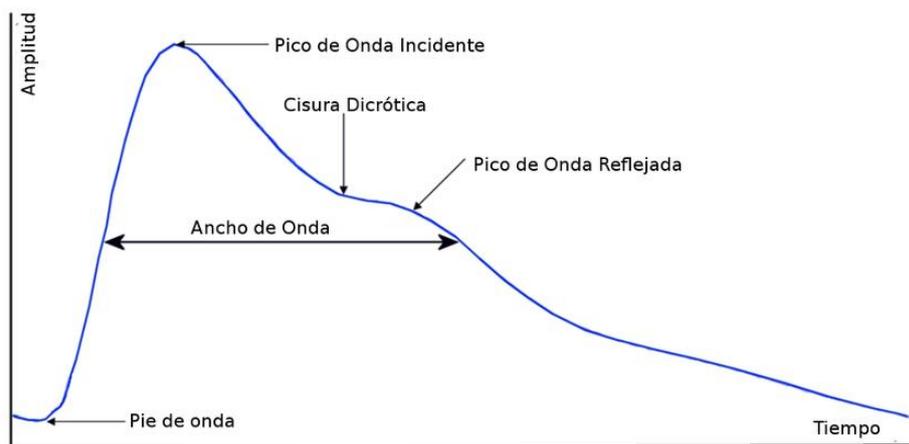
$$Base = \frac{\left(\frac{ciclo}{TotalValores}\right)}{1000}$$

(4)

Por tanto, el área final será la base por la altura. Y la suma de todas las áreas que se calculan en este tramo de la gráfica nos dará el área bajo la curva por cada ciclo respiratorio.

### Cálculo de la cisura dicrótica

La cisura dicrótica es la zona donde se pasa de sístole a diástole. La sístole es la fase de contracción del corazón, donde la sangre es bombeada a los vasos, y la diástole es la fase de relajación, que permite que la sangre entre en el corazón.



En la Figura 14 se muestra una onda donde se muestra la cisura dicrótica

Para el cálculo nos interesa saber en qué posición de la gráfica se encuentra la cisura. Para ello lo que hice es calcular el pico de la cisura dicrótica, es decir, el máximo relativo de la cisura. Usando la misma función del cálculo del PVi tiene una parte donde sabemos, por los valores de las ventanas, que la gráfica está bajando, por tanto, ahí es donde debemos recoger los datos para el cálculo del pico de la cisura.

Una vez obtenido, solo se calcula la diferencia entre este punto y el máximo de la gráfica y la diferencia entre este y el mínimo.

Según la diferencia nos indica si el punto está más cerca del máximo o del mínimo. Esto nos es muy útil para saber si un paciente tiene vasoconstricción o vasodilatación.

Se le denomina vasoconstricción o vasodilatación al aumento o disminución del diámetro de un vaso sanguíneo. Este valor puede variar según factores metabólicos, nerviosos o humorales.

Tono Vascular	Normal	Vasoconstricción		Vasodilatación		
		Severa	Moderada	Leve	Moderada	Severa
Forma Onda						
Amplitud	=	↓↓↓	↓↓	↑	↑↑	↑↑↑
Posición muesca	=	↑↑↑	↑↑	↓	↓↓	↓↓↓

En la Figura 15 se muestra los diferentes tipos de vasoconstricción o vasodilatación según el valor del pico de la cisura dicrotica

En la anterior figura se muestra como lucen la forma de la onda según el tipo de tono vascular, es decir, según el nivel de vasoconstricción o vasodilatación.

# Capítulo 5. Testeo

## 5.1 Testeo conmigo mismo

Para la primera fase del testeo hice unas pruebas conmigo mismo para comprobar los funcionamientos de todas las funcionalidades implementadas.

En la siguiente imagen se muestra una captura tomada en plena ejecución del programa donde se puede ver la gráfica y los valores en ese momento.



En la Figura 16 se puede ver la aplicación en funcionamiento haciendo testeo.

Se puede comprobar, por ejemplo, que según el diseño que se ve de la gráfica el valor del pico en la cisura dicrótica sí que correspondería con una vasodilatación o no presencia de anemia.

Al finalizar la medición me he descargado los datos en Excel y se puede ver lo siguiente:

En el Excel con los datos clínicos:

	A	B	C	D	E	F	G	H	I
1	SpO2	PR/min	RR/min	PI (%)	PVi (%)	Area bajo curva (mm²)	Valor cisuira dicrótica	Pendiente Izquierda	Pendiente Derecha
2		99	62	10	7009	8,641975309	26,29126214 No anemia	2,50632906	12,07317066
3		99	62	10	7118	8,641975309	26,29126214 No anemia	2,200000048	13,6455698
4		99	62	10	7118	8,641975309	26,29126214 No anemia	2,120481968	12,92500019
5		99	60	10	7251	8,641975309	26,29126214 No anemia	2,227848053	11,66265106
6		99	60	10	7251	9,638554217	31,03921569 No anemia	2,550724745	12,39436626
7		99	60	10	7769	9,638554217	31,03921569 No anemia	2,444444418	11,91666698
8		99	60	10	7769	9,638554217	31,03921569 No anemia	2,138888836	13,01408482
9		98	62	10	8296	9,638554217	31,03921569 No anemia	2,838709593	12,75362301
10		98	62	10	8296	4,166666667	23,1804878 No anemia	2,40625	12,73684216

En la Figura 17 se puede ver el Excel con los datos clínicos resultado de la prueba de testeo realizada.

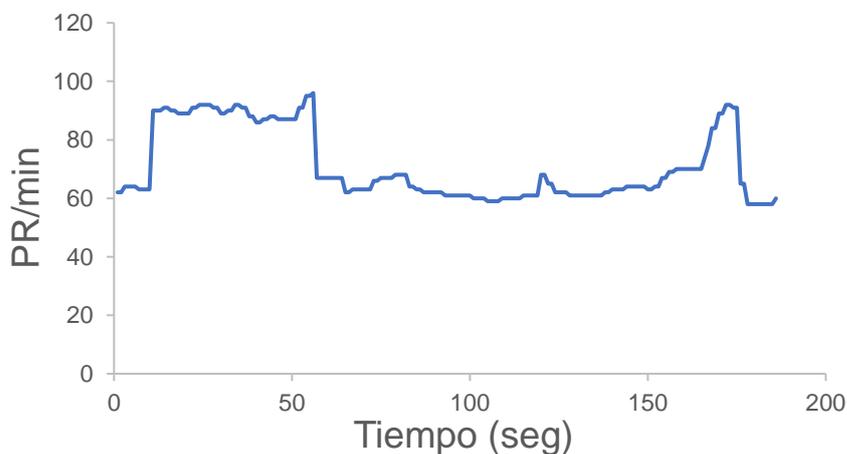
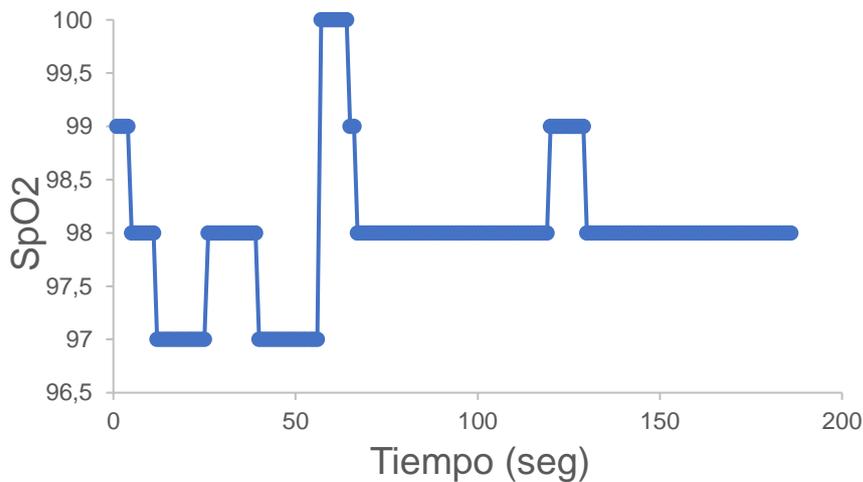
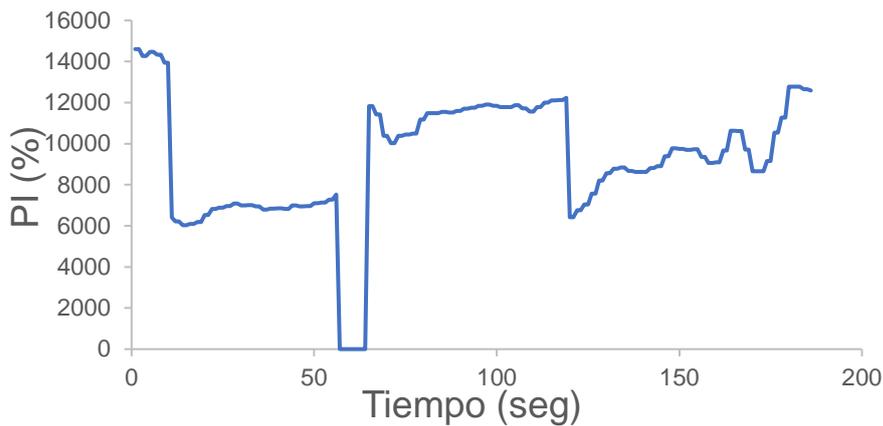
En el Excel con los datos de la gráfica:

	A	B	C	D	E	F	G
1	Grafica						
2	16						
3	13						
4	11						
5	11						
6	11						
7	11						
8	10						
9	6						
10	3						
11	2						
12	0						
13	0						
14	0						
15	0						
16	-1						
17	-3						
18	-8						
19	-11						
20	-13						
21	-16						
22	-21						
23	-26						
24	-29						
25	-31						
26	-34						
27	-39						
28	-40						
29	-37						
30	-30						
31	-13						
32	5						
33	14						
34	20						
35	23						
36	25						
37	26						
38	27						

En la figura 18 se puede ver el Excel con los datos recogidos de la gráfica.

Se pueden llegar a mostrar en las primeras mediciones los valores de  $PVi = 0$ , Área bajo la curva = 0 y el Valor cisura dicrótica = "Sin valor exacto" porque el pulsioxímetro aún se está configurando y se ha programado para que en vez de que se muestren caracteres sin sentido muestre un texto o número más entendible.

Una de las mayores utilidades de obtener los Excel es el posterior tratamiento de los datos que se pueden hacer, por ejemplo, con los datos anteriores se pueden obtener las siguientes gráficas, cada una correspondiendo con un parámetro diferente.



Por tanto, en caso de que sea necesario, se podría hacer un gran estudio de cada paciente a través de sus datos, mostrando gráficas, comparaciones...

## 5.2 Testeo con personas ajenas

Para el testeo tanto de la usabilidad como del funcionamiento he escogido a 3 personas. Una de esas personas es enfermero y por tanto nos podrá dar una información más detallada sobre el funcionamiento y si realmente fuera útil en su puesto de trabajo. Otra persona es una persona totalmente ajena al mundo de la medicina y del mundo de la medicina. Así podré ver si realmente la aplicación es fácil de usar para que una persona sin muchos conocimientos de tecnología ni de medicina puede usar la aplicación sin problemas. Por último, se le dará la aplicación a otra persona que tiene conocimientos de tecnología, pero pocos conocimientos de medicina. Así podré ver como usa la aplicación una persona que entiende de tecnología.

Para las pruebas se les dio a las personas tanto la aplicación como el pulsioxímetro y se les dijo que lo debían usar como ellos creyeran sin recibir ayuda mía ni de nadie externo.

De estas pruebas se sacaron las siguientes conclusiones:

### **Persona con conocimientos médicos**

Esta persona tuvo más facilidades para hacer funcionar el pulsioxímetro en cuanto a enchufarlo y colocación del dedo, cuando entró a la aplicación se le hizo un poco más difícil de lo normal el aceptar todos los permisos necesarios, pero al final lo supo hacer. Después, todo el transcurso de la prueba se hizo según lo esperado, es decir, siguió el flujo de la aplicación tal cual se diseñó. Escaneó el pulsioxímetro, se conectó, visualizó los datos y la gráfica y finalmente guardó sus datos correctamente. Por tanto, excepto el momento de aceptar los permisos, el desarrollo de la prueba con esta persona fue aceptable.

### **Persona sin conocimientos informáticos ni médicos**

Esta persona tuvo un comportamiento ante la aplicación bueno, pero algo lento, es decir, supo cómo hacer funcionar la aplicación correctamente según lo esperado, pero con lentitud. Sobre todo, le costó el hecho de saber que solo con desconectar el pulsioxímetro ya se paran de grabar datos y te lleva automáticamente a la página siguiente, cosa que me hizo pensar de que tal vez sería buena idea poner alguna clase de botón o algo que sea más visual y que con solo pulsarlo pare de grabar los datos y te lleve a la siguiente página. También me comentó que le hubiera gustado que antes de empezar a usar la aplicación se le explicara un poco en que va a consistir y un poco como va a funcionar y como usarla para sacar el máximo de beneficio. Por tanto, el desarrollo de la prueba fue aceptable, pero con ciertas cosas a tener en cuenta en futuras modificaciones.

### **Persona con conocimientos informáticos**

Esta persona tuvo algunas dificultades para hacer funcionar el pulsioxímetro ya que nunca había usado uno y se le hizo algo difícil el enchufarlo y hacerlo funcionar. Pero una vez lo consiguió ya todo fue según lo previsto. Este sí que fue más curioso y antes de empezar a escanear entró a la página de consejos para ver que le decían. Después, el desarrollo por las demás partes de la aplicación fue según lo previsto. Me dijo que le gustaba mucho el diseño que la aplicación y que la forma de mostrar los datos le parecía muy visual, sobre todo para él que no entendía el significado de la mayoría de los parámetros que se mostraban. Por tanto, el desarrollo de la prueba fue muy buena a excepción de la parte del pulsioxímetro.

## Capítulo 6. Manuales

### 6.1 Manual de instalación

Para la instalación del proyecto, solo es necesario la descarga de la rama main del GitHub [9] y abrir esa carpeta en Android Studio. Para instalar solamente la aplicación, lo único necesario será una .apk el cual se entregará y bastará solo con ejecutarlo y él solo se instalará en vuestro dispositivo.

### 6.2 Guía de uso

1. Al iniciar la aplicación se verá una pantalla de carga con el eslogan de la aplicación. Esta página te redirigirá automáticamente a la página principal.



*En la Figura 19 se ve la Splash Screen de la aplicación*

2. En la página principal encontraremos varios botones. Uno de ellos es el de iniciar el escaneo para buscar dispositivos bluetooth cercanos. El otro es un botón flotante el cual si pulsamos se despliegan 2 funcionalidades más. Una de ellas es la de ir a la lista de pacientes y otra la de ir a la página de ayuda/consejos.



En la Figura 20 se puede ver la página principal de la App

- Si pulsamos en lista de pacientes nos llevará a una página donde veremos todos los pacientes que hayamos guardado sus datos. Están ordenados por orden alfabético. Cuando tengamos ya muchos pacientes registrados, puede ser muy engorroso el tener que ir buscando uno a uno, por tanto, hay un desplegable que si lo pulsamos se nos abrirá unas opciones para filtrar los pacientes según fecha, nombre o ambas. Hay dos tipos de Excel, con parámetros clínicos y otro con los datos de la gráfica. Estos se distinguen por un símbolo de un documento o una gráfica.



En la Figura 21 se puede ver la página del historial de pacientes de la App

4. Si desde la página principal pulsamos sobre consejos, nos llevará a una página donde nos darán unos pequeños consejos de uso de la aplicación, así como que son los datos que obtenemos y entre qué rango de valores se consideran correctos esos parámetros. Esto está pensado más para la gente sin conocimientos médicos para que pueda usar la aplicación y entender que está viendo.



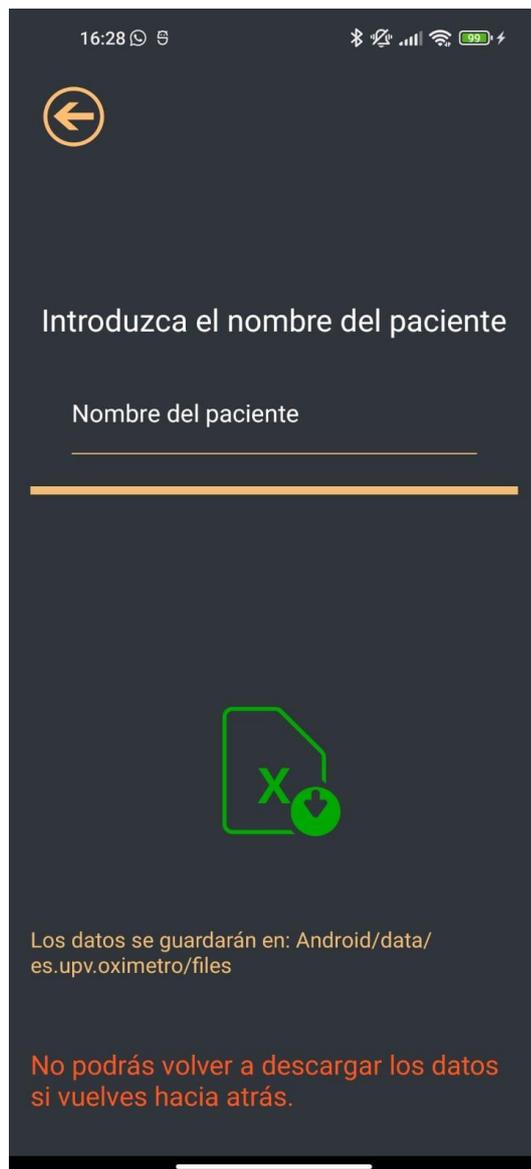
En la Figura 22 se puede ver la página de consejos de la App

5. Una vez escaneado, encontrado nuestro dispositivo y conectado llegaremos a una página donde se visualizarán los datos y la gráfica que se reciben del pulsioxímetro.



En la Figura 23 se puede ver la pantalla donde se muestran los datos.

6. Cuando acabemos de medir los datos, solo con desconectar el pulsioxímetro, a los 5 segundos nos redirigirá automáticamente a la página para guardar el paciente, si así se desea. En esta solo tendremos que poner el nombre del paciente y pulsar en el botón para guardar los datos de este.



*En la Figura 24 se puede ver la pantalla para descargar los datos del paciente.*

7. Finalmente, cuando ya hayamos guardado los datos volveremos a la página principal.
8. Repetir el mismo proceso para cada paciente.

## Capítulo 7. Evaluación

### 7.1 Cumplimiento del objetivo

Durante la realización de este proyecto se han alcanzado todos los objetivos propuestos inicialmente. Se ha investigado el estado actual de este tipo de aplicaciones y se han estudiado estas para después aplicar las mejores funcionalidades y aplicarlas en nuestro proyecto.

Como se explica en el Capítulo “Testeo”, la aplicación ha sido probada por tres personas de distintos perfiles que nos dan una visión diferente sobre el uso de la aplicación. En los tres casos la aplicación ha sido fácil de usar y ninguno se ha quedado trabado en ninguna parte de la aplicación. Todos han conseguido conectar el pulsioxímetro, usarlo, recoger datos y después guardar los datos en Excel.

### 7.2 Conclusiones sobre el proyecto

El poder medir tus ritmos cardíacos desde casa, a un precio muy bajo y de forma muy fácil es una cosa que pocos proyectos han conseguido. Por tanto, se puede decir que este proyecto es un proyecto muy ambicioso, pero con cabida en el mundo y mercado actual y que tiene una utilidad patente.

Este proyecto puede hacer que, a partir de su publicación, en hospitales y centros médicos se pueda empezar a tener conocimientos sobre los pacientes de una manera más rápida, aparte de poder substituir esas máquinas pesadas que se usan ahora para las mediciones por un simple pulsioxímetro y un móvil/Tablet. Es decir, se ganaría en tiempo, cosa que podría hacer que se redujeran las colas en los centros hospitalarios, y en salud, ya que al ser muy rápida la forma de uso y de mediciones hace que los médicos o enfermeros puedan tener una primera visión del paciente mucho más rápido y, por tanto, en caso de poder tener algo grave, poder actuar con antelación.

También el uso doméstico que la aplicación tiene ayuda mucho a que personas, desde sus propias casas, puedan tener un mínimo control sobre ellos y en caso de alguna anomalía poder contactar con su médico para realizar ya una exploración más exhaustiva. Esto es muy útil porque en la actual pandemia del COVID-19 muchos estudios afirman que con solamente un pulsioxímetro en casa se hubieran podido prevenir muchas hospitalizaciones y muchas muertes, ya que uno de los síntomas que más reportaban las personas contagiadas con COVID

era la falta de oxígeno [22] , y por tanto con nuestra aplicación, esto se podría mirar y estudiar. [1]

## 7.3 Aportaciones personales

A nivel personal me ha aportado mucho, a la par que ha sido todo un reto el tener que hacerlo yo solo.

Empezando porque la aplicación cuando me la dieron ya estaba empezada y que por tanto alguien con diferente estilo de programación y diseño ya había estado ahí haciendo trabajo. Esto para cualquier programador es un suplicio porque hasta que empiezas a entender el código y cómo funciona pueden pasar semanas, incluso meses de solo trabajo de entender el código. Por tanto, esto me ha enseñado a cómo afrontar situaciones así y a cómo hacer una buena ingeniería inversa del código para poder entenderlo lo antes posible.

Posteriormente, al realizar este proyecto me ha hecho mejorar mucho mis dotes como programador en Android y Java Android usando Android Studio, y gracias a este proyecto he alcanzado un nivel de programación muy alto en este entorno y lenguajes. También me ha ayudado a mejorar mi capacidad de planificación a la hora de enfrentarme a proyectos así y a saber planear qué tareas debo hacer en qué periodo de tiempo y a planificarlo para llegar a tiempo a la entrega.

Al ser un proyecto cuyos objetivos principales están fuera de mi círculo de estudio o de conocimiento, me supuso un reto el entender qué parámetros nos daba el pulsioxímetro, qué eran, para qué servían etc. Por eso muchos de los problemas de implementación que he tenido han sido dados por eso. Intentaba hacer cosas sin saber qué eran los datos que estaba usando y hasta que comprendí esto cada mejora que quería hacer en este aspecto se demoraba más de lo que debería. Un buen ejemplo es cuando tuve que calcular el índice de variabilidad plétismográfica (PVi). Para esto tuve que entender cómo se mostraba la gráfica del ritmo cardíaco, qué valores son los que se muestran y demás. Y ya cuando supe esto, fue cuando ya me tuve que meter en el código para ver cómo plasmaba todos los conocimientos que había adquirido en la forma de programar el cálculo del PVi, cosa que no resulto nada fácil.

Como conclusión puedo decir que es un proyecto que me ha hecho mejorar mucho en el aspecto técnico de programación, pero que también me ha hecho aprender cosas nuevas a las que no daba importancia en mi vida pero que me he dado cuenta de que es muy importante conocerlas mínimamente. También me ha hecho enfrentarme a retos de programación en los que nunca había estado y que me ha hecho estar semanas e incluso meses pensando en formas diferentes de resolverlos.

## 7.4 Futuras líneas de trabajo

Pese a que se han cumplido los objetivos este es un proyecto que se puede expandir mucho más y se puede mejorar todo lo que queramos. Por ejemplo, se podría hacer una versión de este proyecto para IOS, es decir, para iPhone. Así ampliaría más el espectro en el mercado de posibles clientes.

También se podría expandir la aplicación para que fuera compatible con más tipos de pulsioxímetros aparte de este. Así cada uno podría usar el pulsioxímetro que le pareciera más bonito, cómodo o económico, pero con la misma aplicación para todos.

Guardar los datos en un servidor remoto ofrecería una mejor interconexión de los datos y ofrecería también la posibilidad de que cualquier médico sin importar dónde esté pueda tener acceso a todos los datos de sus pacientes en cualquier momento.

Como conclusión, es un proyecto en el cual se han cumplido los objetivos propuestos pero que en un futuro se podría seguir trabajando en este y mejorándolo para hacerlo aún más competitivo y útil en la vida de las personas.

## Capítulo 8. Referencias

- «Covid-19: cómo funciona el pulsioxímetro y por qué es vital para evitar muertes,» [En línea]. Available: <https://andina.pe/agencia/noticia-coronavirus-como-funciona-pulsioximetro-y-por-es-vital-para-evitar-muertes-video-828963.aspx>. [Último acceso: Mayo 2022].
- [2] «Repositorio Riunet,» [En línea]. Available: <https://riunet.upv.es/handle/10251/11261>. [Último acceso: Mayo 2022].
- [3] D. L. Serrano, «TFG DISEÑO E IMPLEMENTACIÓN DE UN PULSIOXÍMETRO,» [En línea]. Available: <https://riunet.upv.es/bitstream/handle/10251/91753/L%c3%93PEZ%20-%20Dise%c3%b1o%20e%20implementaci%c3%b3n%20de%20un%20pulsiox%c3%admetro.pdf?sequence=1&isAllowed=y>. [Último acceso: Junio 2022].
- [4] C. P. Cruz, «Desarrollo de un sistema IoT integrado con dispositivos eHealth para la detección automática de la variabilidad cardíaca,» [En línea]. Available: <https://riunet.upv.es/bitstream/handle/10251/91750/PALAO%20-%20Desarrollo%20de%20un%20sistema%20IoT%20integrado%20con%20dispositivos%20de%20eHealth%20para%20la%20detecci%c3%b3n%20auto....pdf?sequence=1&isAllowed=y>. [Último acceso: Junio 2022].
- [5] «Masimo Set,» [En línea]. Available: <https://www.masimo.es/technology/co-oximetry/pvi/>. [Último acceso: Mayo 2022].
- [6] M. A. Palazuelo, «APLICACIÓN MÓVIL PARA EL PARSEO Y LA VISUALIZACIÓN DE LOS DATOS DE UN PULSIOXÍMETRO,» 2017-2018. [En línea]. Available: [https://addi.ehu.es/bitstream/handle/10810/29204/TFG\\_PalazueloAingeru.pdf?sequence=1&isAllowed=y](https://addi.ehu.es/bitstream/handle/10810/29204/TFG_PalazueloAingeru.pdf?sequence=1&isAllowed=y). [Último acceso: Mayo 2022].
- [7] «Repositorio Git Hub del TFG de Aingeru Sanchez,» [En línea]. Available: <https://github.com/aingerusanchez/HealthyBeat>. [Último acceso: Mayo 2022].
- [8] «Documento web donde se ven las diferentes convocatorias,» [En línea]. Available: <http://fisabio.san.gva.es/documents/10157/74c202c1-e616-4b2c-8aaf-e571474eba70>. [Último acceso: Junio 2022].
- [9] A. V. Perez, «Repositorio GitHub del TFG Oximeter APP,» [En línea]. Available: <https://github.com/AdrianVaello/TFG-Oximetro>. [Último acceso: Mayo 2022].
- [10] «Sitio de compra del Pulsioxímetro,» [En línea]. Available: <https://www.bdshop.com/finger-clip-pulse-oximeter-fro-200>. [Último acceso: Mayo 2022].

- [11] «Aplicación BLE Scanner,» [En línea]. Available: <https://play.google.com/store/apps/details?id=com.macdom.ble.blescanner&hl=es&gl=US>. [Último acceso: Marzo 2022].
- [12] «BluetoothLeGatt,» [En línea]. Available: <https://github.com/dustedrob/BluetoothLeGatt>. [Último acceso: Marzo 2022].
- [13] «FastBle,» [En línea]. Available: <https://github.com/Jasonchenlijian/FastBle>. [Último acceso: Marzo 2022].
- [14] J. T. Girones, «Repositorio GitHub de la primera versión del proyecto,» [En línea]. Available: <https://github.com/jesus-tomas-girones/Oximetro>. [Último acceso: Marzo 2022].
- [15] «Librería HSSF,» [En línea]. Available: <https://poi.apache.org/apidocs/dev/org/apache/poi/xssf/usermodel/XSSFWorkbook.html>. [Último acceso: Abril 2022].
- [16] «Repositorio de la librería Android Chart,» [En línea]. Available: <https://github.com/PhilJay/MPAndroidChart>. [Último acceso: Marzo 2022].
- [17] A. developer, «Página información File Provider,» [En línea]. Available: <https://developer.android.com/reference/androidx/core/content/FileProvider>. [Último acceso: Mayo 2022].
- [18] R. M. s. Anestesiología, «Información sobre el PVI,» [En línea]. Available: <https://www.medigraphic.com/pdfs/rma/cma-2017/cmas171cw.pdf>. [Último acceso: Mayo 2022].
- [19] «Información de Masimo sobre el PVI,» [En línea]. Available: <https://www.masimo.es/technology/co-oximetry/pvi/>. [Último acceso: Mayo 2022].
- [20] «Calculadora de complemento a dos,» [En línea]. Available: <https://www.allmath.com/es/calculadora-de-complemento-a-dos.php>. [Último acceso: Mayo 2022].
- [21] «MathWorks Calculo de areas,» [En línea]. Available: [https://es.mathworks.com/matlabcentral/fileexchange/105740-calculus-integrals?s\\_v1=43977&elqem=3724449\\_EM\\_ES\\_DIR\\_22-06\\_MOE-EDU](https://es.mathworks.com/matlabcentral/fileexchange/105740-calculus-integrals?s_v1=43977&elqem=3724449_EM_ES_DIR_22-06_MOE-EDU). [Último acceso: 06 2022].
- [22] «Qué pasa con nuestro oxígeno vs la Covid-19? El pulsioxímetro,» [En línea]. Available: <https://www.arpamedica.es/que-pasa-con-nuestro-oxigeno-vs-la-covid-19-el-pulsioximetro/>. [Último acceso: Mayo 2022].
- [23] «Pulsioxímetro para uso domiciliario en pacientes con COVID-19 catalogados inicialmente como casos leves y con factores de riesgo,» [En línea]. Available: <https://sites.bvsalud.org/redetsa/brisa/resource/?id=biblioref.referencesource.1104045>. [Último acceso: Mayo 2022].
- [24] «Polisabio UPV,» [En línea]. Available: <https://www.polisabio.es/index.php/es/>. [Último acceso: Junio 2022].

