



Universidad
Zaragoza

Trabajo Fin de Grado

Desarrollo de una aplicación para la gestión de ritmos de carrera

Development of an application for the management of race rhythms

Autor

Ismael Razquin Plano

Director

Pedro Javier Álvarez Pérez-Aradros

Titulación del autor

Graduado en Ingeniería de Tecnologías Industriales

Eina

Año
2022

RESUMEN

El progreso tecnológico actual ha llevado a la popularización e instalación masiva de aplicaciones en relojes inteligentes. Uno de los usos más habituales son los enfocados a la gestión de actividades deportivas, en concreto al atletismo y running. Ambas disciplinas cuentan con una gran cantidad de usuarios, tanto a nivel profesional como amateur, de forma continuada o esporádica. En ambos casos, es muy importante para el corredor poder gestionar los ritmos de carrera durante su entrenamiento.

Actualmente existen distintas soluciones para controlar el ritmo de carrera al que estamos yendo, tanto con aplicaciones de dispositivos móviles como aplicaciones de relojes. Sin embargo, en la mayoría se presenta el problema de tener que estar continuamente desviando la mirada y la atención del trayecto para conocer el ritmo al que se encuentran; inconveniente que se agrava cuando se trata de corredores con un nivel de exigencia elevado, donde la necesidad de conocer sus ritmos de carrera es permanentemente.

En este Trabajo de Fin de Grado se da solución al problema descrito, desarrollando una aplicación para reloj que permite el seguimiento del ritmo del corredor disminuyendo las distracciones durante el entrenamiento. Para ello, se implementarán señales acústicas y visuales que permitan al corredor, mediante un vistazo rápido obtener información del ritmo al que se encuentra.

En el proyecto podemos destacar las siguientes fases:

- Estudio de mercado para la elección de un dispositivo que permita la instalación de aplicaciones externas.
- Elección del entorno de desarrollo para la escritura del código.
- Desarrollo del código en Android Studio
- Pruebas de funcionamiento, inicialmente en emuladores y posteriormente en dispositivos reales.

(Este documento debe remitirse a seceina@unizar.es dentro del plazo de depósito)

D./D^a. ,

en aplicación de lo dispuesto en el art. 14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de Estudios de la titulación de

(Título del Trabajo)

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza,

Fdo:

ÍNDICE

Resumen	3
Índice	5
Índice de figuras	7
Índice de tablas.....	8
1. Introducción	1
1.1. Contexto y motivación	1
1.2. Objetivos y alcance	2
1.3. Herramientas utilizadas	2
1.3.1. Herramientas de desarrollo y lenguajes de programación	2
1.3.2. Hardware	3
1.3.3. Herramientas de documentación.....	3
1.4. Estructura del proyecto	3
2. Análisis del Sistema	5
2.1. Análisis de requisitos	5
2.1.1. Requisitos del dispositivo	5
2.1.2. Requisitos de la aplicación	7
2.2. Análisis de dispositivos y tecnologías	8
2.2.1. Análisis de dispositivos	9
2.2.2. Análisis de tecnologías para el desarrollo en instalación de la aplicación	13
3. Diseño de la solución.....	15
3.1. Entorno	15
3.2. Estructura de la aplicación.....	16
3.2.1. Configuración.....	17
3.2.2. Gestión de ritmos	18
3.2.3. Detección de alertas	22
3.2.4. Historial de actividades	24
3.3. Diagrama de navegación de la aplicación.....	25
3.3.1. Descripción diagrama de navegación.....	26
3.4. Estudio de usabilidad	30
3.5. Definición de ensayos a realizar	34
4. Experimentos.....	37
4.1. Experimentos en el emulador.....	37
4.2. Experimentos en el reloj	41

5.	Gestión del proyecto	47
5.1.	Organización del proyecto	47
5.2.	Fases del proyecto	47
6.	Conclusiones y trabajo futuro	49
6.1.	Conclusiones del proyecto	49
6.2.	Conocimientos adquiridos	49
6.2.1.	Conocimientos Técnicos	49
6.2.2.	Conocimientos personales	50
6.3.	Dificultades durante la realización del trabajo	51
6.4.	Líneas de trabajo futuras	52
7.	Bibliografía.....	53

ÍNDICE DE FIGURAS

Figura 1. Samsung Galaxy watch 3	11
Figura 2. Motorola moto 360	11
Figura 3. Reloj seleccionado	12
Figura 4. Entorno de la aplicación	15
Figura 5. Estructura de la aplicación	16
Figura 7. Vector de ubicaciones.....	19
Figura 8. Parámetros 1 del vector	19
Figura 9. Parámetros 2 del vector	20
Figura 10. Computación ritmo instantáneo.....	20
Figura 11. Esquema cálculo ritmo medio	21
Figura 12. Esquema funcionamiento señales	22
Figura 6. Esquema de funcionamiento del tiempo entre alertas.....	23
Figura 13. Diagrama de navegación.....	25
Figura 14. Menú de inicio	26
Figura 15. Inicio actividad.....	26
Figura 16. Ritmo objetivo	27
Figura 17. Actividad con ritmo inferior.....	27
Figura 18. Actividad con ritmo superior	27
Figura 19. Actividad principal.....	28
Figura 20. Pantalla de ritmo medio.....	29
Figura 21. Pantalla previa historial de actividades.....	29
Figura 22. Historial de actividades	30
Figura 23. Facilidad de uso de la aplicación	31
Figura 24. Diseño de la interfaz.....	32
Figura 25. Adecuación señales de alerta	32
Figura 26. Mejora de la aplicación con las señales de alerta	33
Figura 27. Satisfacción general con la aplicación.....	34
Figura 28. Primera prueba.....	38
Figura 29. Segunda prueba.....	38
Figura 30. Tercera prueba actividad 1.....	39
Figura 31. Tercera prueba actividad 2.....	39
Figura 32. Cuarta prueba.....	40
Figura 33. Petición de permisos.....	41
Figura 34. Pantalla de inicio	41
Figura 35. Configuración distancia	42
Figura 36. Configuración de ritmo.....	42
Figura 37. Pantalla de configuración.....	42
Figura 38. Ritmo inferior.....	43
Figura 39. Ritmo objetivo	43
Figura 40. Ritmo superior	43
Figura 41. Historial de actividades	43
Figura 42. Actividad reloj.....	44
Figura 43. Actividad móvil	44
Figura 44. Actividad móvil con pausa 2.....	45
Figura 45. Actividad móvil con pausa 1.....	45
Figura 46. Actividad con pausa reloj	46
Figura 47. Distribución horas proyecto	48

ÍNDICE DE TABLAS

<i>Tabla 1. Tabla comparativa de relojes</i>	<i>10</i>
--	-----------

1. INTRODUCCIÓN

En este capítulo se expone el contexto, la motivación y los objetivos de este trabajo, así como las herramientas que se han utilizado durante su desarrollo. Además de ello, comenta cómo se ha estructurado la memoria de este proyecto.

1.1. CONTEXTO Y MOTIVACIÓN

Actualmente una gran cantidad de personas dedican tiempo a la realización de actividades deportivas. Una de las actividades deportivas más de moda por su adecuación a todo tipo de niveles de deportistas y accesibilidad a todos los usuarios es la práctica de running, ya sea como actividad deportiva principal o como actividad complementaria de preparación para la práctica de otros deportes. La mayoría de los corredores, indistintamente del nivel en el que se esté corriendo, emplean aplicaciones de control de ritmos, que permiten el seguimiento de la ruta, la obtención de tanto los ritmos medios como los ritmos instantáneos y la comparación de los distintos trayectos que se van realizando a lo largo de los días.

El problema surge, enfocándonos en los corredores experimentados, que aspectos como la comodidad durante el trayecto y la concentración son vitales para la obtención de buenos resultados durante el entrenamiento, ya que normalmente estos entrenamientos suelen ir enfocados a la mejora de las marcas personales o a conseguir superar barreras anteriores. Estos corredores normalmente emplean dispositivos alternativos a los móviles durante sus actividades, por temas ergonómicos, debido a su gran tamaño y el peso que conlleva correr con ellos, es por ello que suelen optar por el uso de dispositivos que se adapten mejor a las carreras, como los relojes inteligentes. El inconveniente, debido a la ausencia de señales en los relojes que informen al corredor de los ritmos en tiempo real, es que se obliga al corredor a enfocar su atención en el dispositivo.

Es por ello que este proyecto se centra en el desarrollo de una aplicación que, mediante señales, permita al corredor saber en qué rango de velocidad se encuentra corriendo. Para ello se emplearán tanto señales auditivas como visuales cuando el corredor se aleje del ritmo establecido para el trayecto. De esta forma, se pretende que el usuario no tenga que estar mirando el reloj cada vez que quiera conocer si está yendo dentro del intervalo de velocidad que se había propuesto, ya que, si éste corre a velocidades fuera del mismo, el dispositivo le avisa mediante las señales implementadas.

1.2. OBJETIVOS Y ALCANCE

Este Trabajo de Fin de Grado tiene el objetivo principal de desarrollar una aplicación funcional para relojes inteligentes que permita el control de los ritmos de carrera del usuario. Este objetivo puede desglosarse en objetivos más detallados como:

- Sistema de obtención de ubicaciones periódico
- Capacidad de envío de señales de alerta
- Acceso a un histórico de actividades
- Experimentación en dispositivos reales

Queda fuera del alcance del proyecto, el desarrollo de una aplicación capaz de configurarse para distintos tramos de carrera.

1.3. HERRAMIENTAS UTILIZADAS

A continuación, se exponen las herramientas utilizadas durante el desarrollo del proyecto:

1.3.1. Herramientas de desarrollo y lenguajes de programación

- **Android Studio [16] [17]:** Android Studio es el entorno de desarrollo que se empleará durante el proyecto. Es el entorno oficial de Android que permite crear aplicaciones mediante la programación en Java, lenguaje en el cual se encuentran desarrolladas las aplicaciones de los dispositivos de Android y aquellos con sistema operativo Wear OS.

Android Studio ofrece la posibilidad de utilizar emuladores para la realización de pruebas durante el desarrollo del código, de forma que se puedan ir constatando el cumplimiento de los objetivos que se han declarado previamente.

- **Java [30]:** lenguaje de programación utilizado para el desarrollo de la aplicación. Java es el lenguaje que emplea Google para el desarrollo de las aplicaciones.
- **XML [29]:** proporciona una plataforma para definir elementos para crear un formato y generar un lenguaje personalizado. Se ha empleado para el diseño de la interfaz de la aplicación.⁹
- **Gradle [31]:** empleado para la automatización y el proceso de compilación de la aplicación.

1.3.2. Hardware

- **Smart-watch:** dispositivo real en el cual se instalará la aplicación para la realización de pruebas de la aplicación.

Este dispositivo tiene conexión Bluetooth, Wifi y GPS, requeridas para el funcionamiento de la aplicación, así como otros sensores de medición, como medición del ritmo cardiaco, contador de pasos... que pueden dar información al usuario acerca de los trayectos, de forma complementaria a la que proporciona la aplicación desarrollada.

1.3.3. Herramientas de documentación

- **Draw.io:** aplicación web utilizada para el diseño de diagramas y gráficos
- **Google Docs:** herramienta utilizada para la redacción de la memoria y obtención de gráficos.
- **Paint:** herramienta empleada para el diseño de gráficos.

1.4. ESTRUCTURA DEL PROYECTO

Este presente Trabajo de Fin de Grado se ha comenzado mediante una introducción al proyecto que se ha realizado, detallando aspectos tales como el objetivo y alcance para realizar el proyecto y las herramientas que se han utilizado.

En el siguiente capítulo, se ha realizado el análisis del sistema, tanto de la selección de dispositivos como de las tecnologías que se van a emplear durante el desarrollo del proyecto. Además, se describirán los requisitos básicos necesarios, en el software que se vaya a utilizar para la escritura del código y en el hardware, en este caso el reloj que será seleccionado.

Una vez elegido el dispositivo, en el siguiente capítulo se cubre la fase de diseño de la aplicación, tras realizar una breve descripción del entorno de desarrollo empleado y cómo se va a estructurar la aplicación para el cumplimiento de los objetivos relativos a esta. Se realizará también el diseño de la interfaz de forma que se cubran las necesidades supuestas básicas de la aplicación y se adapte de la mejor forma posible al usuario, facilitando su manejo.

En el cuarto capítulo se presenta la realización de experimentos, realizados tanto en el emulador, para comprobar el funcionamiento de la aplicación inicialmente y los experimentos realizados posteriormente en el reloj seleccionado.

Finalmente, en las dos últimas partes de la memoria se expone como se ha gestionado el proyecto, en cuanto a la metodología de desarrollo del mismo y el seguimiento que se ha llevado del plan de trabajo. Se incluye como se han distribuido las horas dedicadas en el proyecto, así como la exposición de las conclusiones finales una vez obtenidos los resultados de este, conclusiones tanto de la experiencia personal, como conclusiones técnicas, analizando las dificultades en el uso de las tecnologías y las dificultades durante la realización del trabajo. Además, se realizarán propuestas para trabajos futuros que puedan continuar con el desarrollo de esta aplicación, introduciendo mejoras y optimizando su funcionamiento, usando este proyecto como base de partida.

2. ANÁLISIS DEL SISTEMA

En este capítulo se analizan los requisitos de cumplimiento básico del dispositivo, así como características que puedan influir en la elección final del mismo. Además, se analizarán las tecnologías utilizadas en el proyecto, tanto el entorno de desarrollo, como el método de instalación de la aplicación.

2.1. ANÁLISIS DE REQUISITOS

En cuanto a los requisitos que presenta el proyecto, se van a analizar desde el punto de vista del dispositivo, con aquellas funciones y características que deba cumplir necesariamente para un funcionamiento óptimo. Se tendrán en cuenta también, los requisitos con los que deba cumplir la aplicación, con las labores que tenga que desempeñar, tanto desde la perspectiva visual como la funcional.

2.1.1. Requisitos del dispositivo

Para la elección del dispositivo y establecer aquellas características básicas con las que debe contar, es muy importante tener en cuenta los principales objetivos del proyecto, que son la obtención de la ubicación y la implementación de mejoras, por lo que debe ser capaz de emitir señales visuales y auditivas.

Es por ello que se fijaron una serie de requisitos básicos que debe cumplir el dispositivo para el funcionamiento de la aplicación:

- **GPS incorporado**
- **Conexión Wi-fi y Bluetooth**
- **Sistema operativo con compatibilidades**
- **Instalación de aplicaciones desarrolladas por terceros**
- **Ergonomía**

GPS incorporado: el primer requisito es que el dispositivo tenga *GPS* incorporado, el cual va a permitir el seguimiento de la posición durante todo el trayecto y de éste dependerá la calidad de la ubicación obtenida, por lo tanto, es interesante que el reloj tenga un GPS con buena precisión de toma de datos.

Conexión Wi-fi y Bluetooth: la *conexión bluetooth* y *Wi-fi*, que permite la conexión entre el Smart-watch y nuestro dispositivo móvil para transmitir la información y la posterior instalación de la aplicación. Además de ello, mediante la conexión Wi-fi aseguramos la posibilidad de conectar a Internet.

Sistema operativo: el *sistema operativo* se seleccionará teniendo en cuenta aquellos que ofrezcan mayores compatibilidades, es decir, permitan la instalación de aplicaciones desarrolladas para Android, iOS o Google.

Instalación de aplicaciones desarrolladas por terceros: el requisito más relevante para elegir el dispositivo, dado que los otros requisitos por norma general suelen cumplirse en los dispositivos modernos, es que permitan *instalar aplicaciones diseñadas por terceros*, es decir, por usuarios o desarrolladores que no pertenecen a Google. Es muy importante prestar especial atención a este criterio ya que actualmente los Smart-watch tienen una serie de aplicaciones preinstaladas en el dispositivo y permiten instalar aplicaciones de bibliotecas predeterminadas, por lo tanto, en la gran mayoría de dispositivos sería imposible instalar aplicaciones desarrolladas por uno mismo.

Ergonomía: para analizar la *ergonomía* del dispositivo se tendrá en cuenta si es un reloj diseñado para el ámbito deportivo o tiene un estilo más clásico. Para ello, principalmente se analizará el material empleado en la correa del reloj, que sea un material flexible y cómodo, que se adapte correctamente a la muñeca del usuario.

Los requisitos descritos anteriormente serían aquellos indispensables en el dispositivo, ya que la falta de los mismos haría imposible el desarrollo de este proyecto.

Una vez descritos estos, se definen aquellos que tienen influencia en la elección del dispositivo, pero no son requisitos de aceptación/rechazo:

- **Duración de la batería**
- **Memoria**
- **Peso**
- **Coste**

Duración de la batería: el criterio de la *duración de la batería* se analizará a través de la capacidad de la batería (mAh) y el consumo que tiene el dispositivo, analizando así la autonomía de la batería, estableciendo un mínimo de dos días de uso normal.

Memoria: la *memoria* es necesaria para soportar el tamaño de instalación de la aplicación y para el almacenamiento de datos en ficheros de texto, a los que podamos acceder, para mostrar al usuario las actividades que ha ido realizando con la aplicación. Se establece como límite inferior de memoria, una memoria de 2Gb.

Peso: en cuanto al *peso*, es importante que el peso del reloj sea pequeño, de forma que no suponga una carga al corredor, dado que la ergonomía y el peso son dos de los principales motivos por los cuales se utilizan los relojes inteligentes por encima de los móviles para las actividades de correr.

Coste: en el *coste* se ha establecido un límite superior de 400€ para la adquisición del dispositivo, intentando seleccionar aquel reloj con mejor calidad/precio.

Se tendrán en cuenta características adicionales como la presencia de sensores, que sean capaces de realizar la medición de los pasos recorridos, las pulsaciones durante el trayecto, simetrías al andar.

2.1.2. Requisitos de la aplicación

Para el análisis de los requisitos que tiene que cumplir la aplicación es importante tener presentes cuáles son los objetivos principales del proyecto, como son la obtención de la ubicación y ritmos de carrera. También se han de tener en cuenta los funcionamientos más básicos de la aplicación y como se produce la interacción con el usuario, de forma que, mediante el cumplimiento de estos, la interacción sea lo más cómoda y sencilla posible para el corredor.

En cuanto al cumplimiento de objetivos, se espera que la aplicación sea capaz de:

- **Obtención de la ubicación**
- **Ritmo de carrera**
- **Envío de señales acústicas y visuales**
- **Optimización de la interfaz**
- **Historial de actividades**
- **Capacidad de configuración**
- **Almacenamiento de datos en ficheros**
- **Capacidad de pausar la actividad**

Obtención de la ubicación: la aplicación debe ser capaz de obtener la ubicación del usuario; parámetro fundamental para el posterior cálculo de los ritmos y las distancias recorridas durante la actividad.

Ritmo de carrera: el ritmo de carrera, obtenido mediante la computación de las ubicaciones y el transcurso del tiempo interno del dispositivo.

Envío de señales acústicas y visuales: la capacidad de enviar señales, tanto auditivas como sonoras al usuario, cuando se encuentre corriendo a ritmos de carrera fuera del intervalo objetivo establecido. Esto viene de la mano del cumplimiento de los objetivos de mejora con respecto a aplicaciones convencionales de correr.

Optimización de la interfaz: enfocándonos en el aspecto visual y de interacción con el usuario, los requisitos básicos están relacionados con que la aplicación permita una visualización correcta de aquellos parámetros a mostrar al usuario. Este aspecto es importante debido al reducido tamaño de pantalla con el que se cuenta, por lo que será importante la optimización de los espacios y la correcta estructuración de la información a mostrar en cada una de las pantallas. Además, es importante que la aplicación este

diseñada de forma que la navegación por la aplicación sea sencilla y usuarios no entendidos en tecnologías sean capaces de sacarle el máximo partido a la aplicación.

Historial de actividades: muestra al usuario un historial de actividades secuencial, con el ritmo, distancia y fecha de las actividades realizadas. Muestra el número de veces que el corredor ha rebasado los límites de ritmo objetivos.

Capacidad de configuración: el usuario tiene la capacidad de configurar el ritmo y la distancia que quiere recorrer en el trayecto.

Almacenamiento de datos en ficheros: permite almacenar datos de las actividades y su acceso para ser mostrados al usuario.

Capacidad de pausar la actividad: la aplicación permite pausar la actividad durante el trayecto, parando la toma de datos de distancias y el tiempo transcurrido entre pausas.

2.2. ANÁLISIS DE DISPOSITIVOS Y TECNOLOGÍAS

Para la realización del proyecto es importante realizar el análisis tanto de los relojes inteligentes disponibles en el mercado, como de las tecnologías que se pueden emplear en el desarrollo del trabajo.

Referido a los dispositivos cabe analizarlos desde el punto de vista del cumplimiento de los requerimientos que se han definido previamente, teniendo en cuenta los requisitos básicos, de aceptación/rechazo y aquellos que tienen cierta influencia en la elección del dispositivo, pero no deberían descartarlos inicialmente.

En cuanto a las tecnologías utilizadas para el desarrollo del proyecto, se escogerán aquellas con mayor flexibilidad, de forma que ofrezcan varias posibilidades a la hora de desarrollar el trabajo. Además, se tendrá en cuenta aquella que permita una instalación sencilla de la aplicación a los relojes, es decir, que tenga compatibilidades con este tipo de dispositivos y tenga un método claro de instalación.

2.2.1. Análisis de dispositivos

Definidos los requisitos básicos que debe cumplir el dispositivo, se realizó una primera selección de dispositivos. Se seleccionaron inicialmente aquellos con una mayor tasa de mercado y popularidad entre los usuarios, además de que permitiesen la instalación de aplicaciones.

- **Samsung Galaxy watch 3**
- **Motorola moto 360 sport**
- **Samsung Galaxy watch active 2**
- **Samsung Galaxy watch 4**
- **Oppo watch**
- **Motorola moto 360**

Para comparar los dispositivos se ha realizado la tabla 1 en la cual se exponen las características de cada uno de ellos, referidas a los criterios comentados:

Tabla 1. Tabla comparativa de relojes

Dispositivos	Batería	Memoria	Peso	S.Operativo	Precio	GPS	Wi-fi/Bluettoth	Aplicaciones por terceros	Ergonomía
Samsung Galaxy watch 3	340mAh	8Gb	53g	Tizen	450 €	Sí	Sí	Permite instalación	Clásica
Motorola moto 360 sport	300mAh	4Gb	54g	Android	300 €	Sí	Sí	Permite instalación	Deportivo
Samsung Galaxy watch active 2	320mAh	4Gb	44g	Tizen	260 €	Sí	Sí	Permite instalación	Deportivo
Samsung Galaxy watch 4	361mAh	16Gb	31g	Wear OS	300 €	Sí	Sí	Permite instalación	Deportivo
Oppo watch	430mAh	8Gb	40g	Wear OS	400 €	Sí	Sí	Permite instalación	Deportivo
Motorola moto 360	355mAh	8Gb	52g	Wear OS	315 €	Sí	Sí	Permite instalación	Clásica

Fuente [3] [11] [12] [13]

De entre los dispositivos seleccionados inicialmente se descartaron dos de ellos por la ergonomía que presentaban, el *Samsung Galaxy watch 3*(figura 1), y el *Motorola Moto 360*(figura 2), que tenían diseños poco deportivos y más orientados a la estética, diseños clásicos.



Figura 1. *Samsung Galaxy watch 3*



Figura 2. *Motorola moto 360*

Una vez descartados dispositivos por ergonomía, el criterio fundamental que se va a emplear para la elección del dispositivo es el sistema operativo. Se va a seleccionar aquel sistema que nos de mayores compatibilidades y facilidades para la instalación de la aplicación, de forma que las características que no son de aceptación/rechazo, se emplearán para la selección final del reloj.

Uno de los sistemas operativos que presentan los dispositivos es el Tizen [8], sistema operativo desarrollado por Samsung. La desventaja de este sistema operativo es las bajas compatibilidades y escasez de información al respecto que ofrece, al no contar con la certificación de Google Play y no tener acceso a muchos servicios de Google, por lo que elementos principales de la aplicación podrían no ser compatibles y no funcionar correctamente. Es por ello que el *Samsung Galaxy watch active 2* se descartó.

El sistema operativo Android [9] está diseñado tanto para dispositivos móviles como para relojes inteligentes. Este sistema ofrece compatibilidades con distintas versiones de Android, Chrome... Además de ello es compatible con Google y sus funciones por lo que este sistema operativo es apto para el desarrollo de la aplicación y debería permitir la instalación de la misma en los dispositivos que estén basados en este sistema.

El último sistema operativo es el Wear OS [7], sistema operativo basado en Android que presentó Google. El sistema en sí está optimizado para el uso en relojes inteligentes. El sistema ofrece compatibilidades con distintas versiones de iOS y Android, por lo que es totalmente apto para el desarrollo del proyecto.

Por ello mediante el análisis de los sistemas operativos, se terminaron seleccionando los dispositivos con sistema operativo Wear OS (*Samsung Galaxy watch 4* y *Oppo watch*), especialmente debido a que dicho sistema está desarrollado por Google y basado en Android. Además, es un sistema operativo desarrollado exclusivamente para el uso en Smart-watch, por lo que el grado de optimización para estos dispositivos será mayor que en el caso de los dispositivos con sistema operativo Android.

Para finalizar la elección, se analizaron las características de no aceptación/rechazo de la tabla 1 respectivas a estos dispositivos. El *Samsung Galaxy watch 4* ofrece mejores prestaciones en cuanto a la memoria y peso del dispositivo y tiene un precio más asequible que el *Oppo watch*, mientras que éste únicamente aventaja al Samsung en la batería. Por ello finalmente se decidió escoger el *Samsung Galaxy watch 4* como dispositivo para la realización de pruebas de la aplicación.



Figura 3. Reloj seleccionado

El dispositivo presentado en la figura 3 es el reloj elegido para el proyecto. Cumple con los requerimientos básicos y presenta un sistema operativo que ofrece las compatibilidades necesarias para el desarrollo del proyecto. Dispone de una gran cantidad de sensores que miden parámetros del corredor, como las pulsaciones, la presión arterial, tu composición corporal en tiempo real, que se acoplan perfectamente a la aplicación desarrollada y que pueden ayudar durante la actividad. Cuenta con una dimensión de 44mm de diámetro, un peso de 30 gramos con correa de silicona que se adaptan perfectamente al corredor, tanto por la forma como por los materiales empleados.

2.2.2. Análisis de tecnologías para el desarrollo en instalación de la aplicación

En cuanto a las tecnologías que se van a utilizar durante el proyecto, cabe analizar el entorno de desarrollo que se va a utilizar para elaborar la aplicación y el método que se ha empleado para la instalación de la aplicación desde el ordenador al reloj.

2.2.2.1. Entornos de desarrollo

Como criterios de análisis se tuvieron en cuenta:

- **Compatibilidades con sistemas operativos**
- **Sistema de generación de archivo APK propio**
- **Optimización a aplicaciones de relojes inteligentes**

Se tuvieron en cuenta inicialmente los entornos de desarrollo más populares para crear aplicaciones, como React-Native [14] [17], Qt Creator [15] [17] o Android Studio.

En el caso de **React-Native** [14] [17], los problemas que se encontraron fueron las compatibilidades a la hora de instalar aplicaciones en entornos de relojes inteligentes y que estaba mayormente enfocada en el desarrollo de código para páginas web. Por lo que se descartó debido a la incertidumbre a la hora de ser funcional para el desarrollo de aplicaciones para relojes inteligentes.

Qt Creator [15] [17] es un editor que permite la escritura del código en distintos lenguajes como C++ o JavaScript, además cuenta con la presencia de emuladores. El problema que se encontró es similar al anterior, ya que no estaba descrita la funcionalidad de las aplicaciones desarrolladas en dispositivos móviles. La aplicación utiliza programas externos para la compilación del código y generación del APK, utilizado para la instalación en el reloj.

El entorno de desarrollo que se eligió finalmente fue **Android Studio** [16] [17], IDE oficial para el desarrollo de aplicaciones Android desde 2013.

Este entorno de desarrollo ofrece una gran cantidad de posibilidades, ya que permite la creación de aplicaciones tanto para reloj como para dispositivos móviles facilitando el desarrollo de aplicaciones iniciales. El entorno de Android Studio no nos permite únicamente el desarrollo de aplicaciones, sino que además nos ofrece un sistema de emuladores que nos permite realizar pruebas en dispositivos ficticios. El sistema de emuladores, facilita enormemente hacer avances en el proyecto, ya que permanentemente se puede estar comprobando si los cambios realizados en el código tienen el funcionamiento esperado.

Otro aspecto a tener en cuenta de Android Studio referido a los emuladores, es que permite tener el control de la ubicación del emulador, de forma que se pueden realizar las pruebas también en el emulador. Estas pruebas son fundamentales, ya que el acceso a la ubicación es una de las partes clave del proyecto.

A demás, a diferencia de Qt Creator, tiene la posibilidad de compilar el código en un APK directamente, lo cual facilita mucho la labor de instalación de la aplicación en el dispositivo.

2.2.2.2. Instalación de la aplicación

En cuanto las tecnologías utilizadas para la instalación de la aplicación en el reloj, el abanico de posibilidades encontrado fue prácticamente nulo, dado que en general las aplicaciones disponibles y métodos de instalación están diseñados para la instalación de aplicaciones predeterminadas. La instalación de estas aplicaciones suele ser a través de softwares oficiales como Play Store [32] o a través de Galaxy Wereable [33], de forma que no estaban enfocadas ni preparadas para la instalación de aplicaciones desarrolladas por terceros. La opción más viable que se encontró fue la utilización de una aplicación móvil llamada Bugjaeger Mobile [18].

Este software, permite instalar un APK a través de un dispositivo móvil al reloj mediante conectividad Wi-fi, uno de los requisitos iniciales con los que debía cumplir el reloj. Inicialmente se debe descargar la aplicación en el dispositivo móvil, procedimiento sencillo mediante la conexión entre el móvil y el ordenador con el APK descargado. Posteriormente, se conecta el reloj con el móvil, mediante la conexión depurada por Wi-fi y se transmite el archivo al reloj, quedando instalada la aplicación en éste.

3. DISEÑO DE LA SOLUCIÓN

En este capítulo se va describir el entorno que tiene la aplicación y el funcionamiento esperado de ésta, así como la forma en la que se ha estructurado. Se expone también el diagrama de navegación de la aplicación, el diseño de la interfaz y la batería de pruebas desarrollada para comprobar el funcionamiento de la aplicación.

3.1. ENTORNO

En este apartado, se describe el entorno en el que se ha ejecutado la aplicación y se explica la utilidad de las mejoras implementadas y su forma de interacción con el usuario.

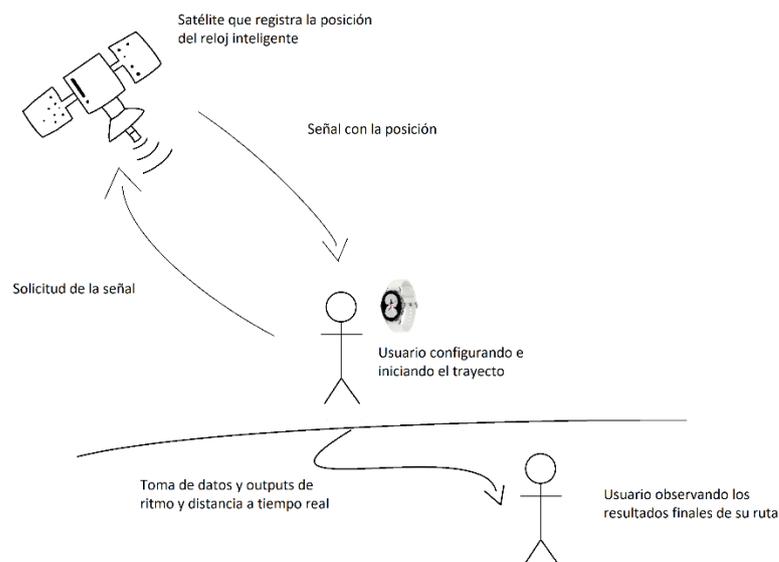


Figura 4. Entorno de la aplicación

En la figura 4, se representa la situación que se quiere abordar y resolver en este Trabajo de Fin de Grado. Refleja a un usuario que sale a realizar una actividad física. El usuario quiere llevar a cabo un seguimiento de la distancia que quiere recorrer y el ritmo al que quiere ir durante su recorrido. Para el control de la distancia y ritmo durante la actividad, el usuario emplea la aplicación desarrollada en este proyecto.

Previo al inicio de la actividad, al usuario se le permite configurar la distancia que quiere recorrer y el ritmo que quiere mantener durante el trayecto. Parámetros que utilizará el

programa durante la actividad. Una vez definidas estas variables, el corredor inicia el trayecto obteniéndose los ritmos de carrera a tiempo real y la distancia total que ha recorrido hasta el momento.

En función del ritmo objetivo y el ritmo al que se encuentre corriendo el usuario, se le enviarán señales informativas que le permitirán conocer de forma sencilla si está yendo al ritmo establecido objetivo o a un ritmo superior o inferior. Una vez finalizada la actividad, bien por recorrer la distancia establecida o por que el usuario finalice el entrenamiento de forma manual, este tendrá acceso a un histórico para revisar tanto la actividad realizada como las previas.

3.2. ESTRUCTURA DE LA APLICACIÓN

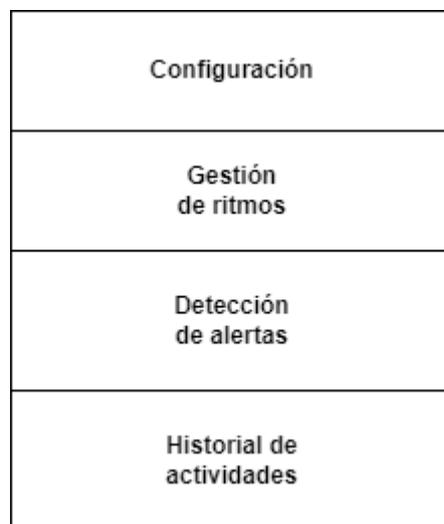


Figura 5. Estructura de la aplicación

La figura 5 muestra como se ha estructurado la aplicación. A cada uno de los bloques, le llegará información de entrada que procesará para la obtención de outputs de utilidad para otros procesos del programa o como información mostrada al propio usuario.

3.2.1. Configuración

La configuración de la aplicación se realiza a través de numerosos parámetros. Algunos de ellos se podrán configurar a nivel de usuario, mientras que otros se han predefinido durante el desarrollo de la aplicación.

A **nivel de usuario** se permite configurar los siguientes parámetros:

- Ritmo objetivo
- Distancia
- Permisos

El ritmo objetivo y la distancia son los parámetros más importantes de configuración. Fijados, se calcula una banda de ritmo válida y si no se cumple se enviarán señales de aviso al usuario, de forma que mediante su configuración se está estableciendo el nivel de exigencia que se impone el usuario durante el trayecto.

El ritmo y la distancia son variables que de forma inicial vienen predefinidas, siendo la distancia inicialmente 5 km y el ritmo objetivo 0:00, de forma que si el usuario no los modifica serán los objetivos durante la actividad. Si el usuario los modifica, durante la actividad se realizarán los cálculos y comparaciones en torno a éstos y serán reseteados a los valores estándar al finalizar la actividad.

Cuando el usuario inicia la aplicación, se le solicita la aceptación del permiso para el acceso a la ubicación. El usuario puede aceptarlos únicamente durante esa actividad, aceptarlos permanentemente o rechazarlos. Si el usuario rechaza estos permisos la aplicación no funcionará correctamente, ya que no se le permitirá obtener la ubicación durante el recorrido.

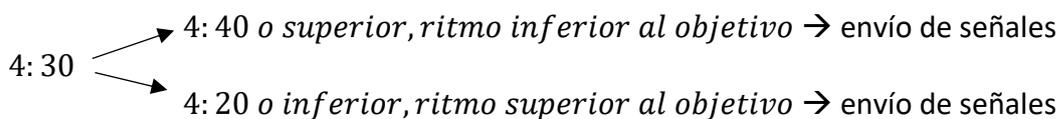
A nivel de programación se han configurado los siguientes parámetros:

- Intervalos de actualización
- Gap
- Intervalo de adaptación
- Tiempo entre alertas

Los intervalos de actualización son los intervalos de tiempo que utiliza la aplicación cuando solicita ubicaciones. Estos intervalos determinan el tiempo que transcurrirá entre la petición de dos ubicaciones en función de la calidad de la señal. El intervalo de actualización por defecto, se ha establecido en 4 segundos, mientras que el intervalo rápido es de 2 segundos.

El Gap es el tiempo utilizado para determinar el intervalo de ritmos límite durante la carrera. Si el usuario supera dicho intervalo, tanto por el límite superior como por el inferior, se envían las señales de alerta.

$Gap = 10 \text{ segundos}, Ritmo objetivo = 4:30$



El intervalo de adaptación es el tiempo que se prevé necesario para que el corredor se adapte al ritmo establecido como objetivo. El intervalo se utiliza de forma que si el tiempo transcurrido es inferior a éste y el usuario está corriendo fuera del intervalo de ritmo objetivo, no se envíen las señales de alerta. Por defecto se establece un intervalo de adaptación de 30 segundos.

El Gap y el intervalo de adaptación son parámetros almacenados en un fichero de propiedades. A los ficheros de propiedades se puede acceder como desarrollador y se puede modificar su contenido, con la ventaja de no tener que reinstalar la aplicación para realizar modificaciones en ésta.

El tiempo entre alertas es un parámetro fijo establecido durante el desarrollo del programa, utilizado como parámetro de comparación. El tiempo entre alertas se ha fijado en 30 segundos.

3.2.2. Gestión de ritmos

El primer paso para la obtención de los ritmos de carrera y las distancias recorridas es la obtención de la ubicación. Es por ello que es interesante conocer el procedimiento de obtención de la misma.

Para la obtención de la ubicación se utilizan diversos sistemas de satélites. La conexión más común se realiza a través del sistema GPS [28], *Global Position System*. El sistema GPS es un conjunto de satélites que orbita a 20.000 km del suelo, para generar señales que cubren todo el planeta. Como se ha comentado existen otras redes de satélites, como la china (*BeiDou*) o la europea (*Galileo*). El uso combinado de todas las redes de satélites existentes, permite al dispositivo obtener con gran precisión su posicionamiento en el mapa.

El dispositivo debe estar conectado a una media de entre 10-15 satélites, para ser capaz de obtener la ubicación de forma precisa. El cálculo de la ubicación se realiza a través de la trilateración de las señales, mediante el trazado de circunferencias, con centro en la señal de los satélites a los cuales se encuentre conectado.

La aplicación que se ha desarrollado, se comunica con el sistema de satélites de forma periódica, adaptándose a la calidad de la señal que es capaz de obtener a través del conjunto de satélites. Si la calidad de la señal es buena, la actualización de la ubicación se realiza con el intervalo de tiempo rápido, mientras que, si la calidad de la señal es más baja, se realizan actualizaciones con el intervalo de tiempo por defecto. La calidad de la señal dependerá del número de satélites con los que nuestro reloj esté estableciendo conexión en cada instante de tiempo.

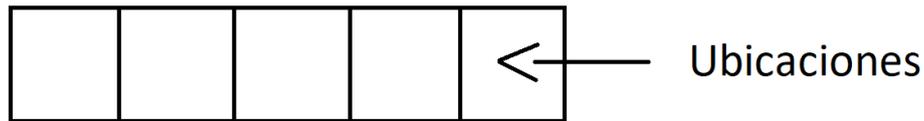


Figura 6. Vector de ubicaciones

Las ubicaciones que se vayan obteniendo se almacenan en el array representado en la figura 7. El array es un vector con capacidad máxima de 5 ubicaciones, que conforme se vayan obteniendo se irán almacenando. La limitación en el número de ubicaciones, es debido a que únicamente se van a guardar las ubicaciones necesarias para el cálculo de ritmos, para evitar la saturación del vector y el almacenamiento de datos inservibles.

Las ubicaciones se almacenan siempre en la primera posición del vector (posición 0), de forma que el resto de ubicaciones se desplazan una posición en el vector. Cuando se han almacenado 5 ubicaciones, al producirse una nueva actualización, la ubicación de la quinta posición del vector es borrada automáticamente.

Las ubicaciones que se obtienen a través de la aplicación vienen dadas en forma de coordenadas geométricas, con parámetros de latitud, longitud, altitud y tiempo. En el desarrollo del proyecto únicamente utilizaremos la longitud y latitud de cada una de las ubicaciones.

Conforme se vayan actualizando las ubicaciones se realizará el cálculo de la distancia recorrida y los ritmos a tiempo real. Para ello serán necesarios los parámetros D_o , T_t y D_t (figuras 8 y 9).

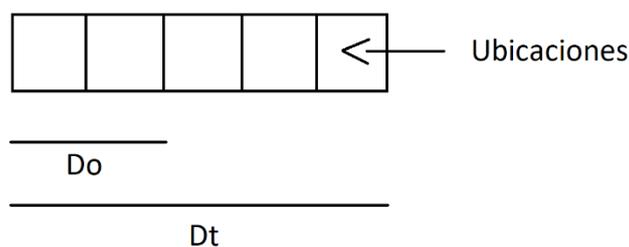


Figura 7. Parámetros 1 del vector

Para el cálculo de la distancia recorrida, únicamente hace falta el parámetro D_o (figura 8), distancia entre la última ubicación obtenida y la ubicación previa. Para realizar el cálculo se utiliza una función propia de Google, que permite la obtención de la distancia entre puntos introduciendo las coordenadas de ambos. Conforme se van actualizando las ubicaciones se va obteniendo la distancia D_o y realizando el sumatorio de la distancia total recorrida.

En el cálculo del ritmo se usarán los parámetros T_t y D_t .

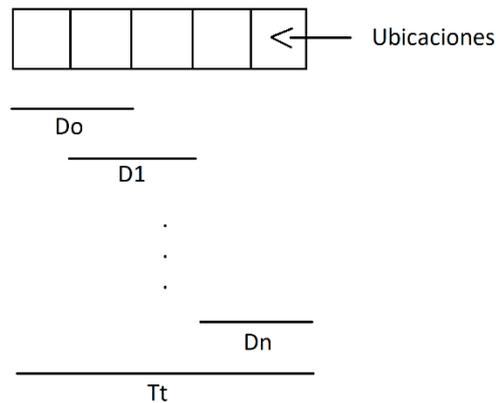


Figura 8. Parámetros 2 del vector

Para la computación del ritmo se emplean las últimas 5 ubicaciones, con el objetivo de evitar grandes oscilaciones si en un determinado momento hay un fallo en la captura de la ubicación o una variación del ritmo momentánea.

Se realiza el sumatorio de las distancias entre las 5 ubicaciones D_t y se calcula el transcurso de tiempo entre la última ubicación obtenida y la ubicación en la última posición del vector, para la obtención de T_t .

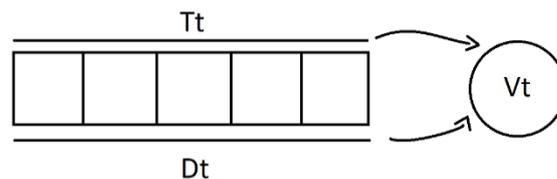


Figura 9. Computación ritmo instantáneo

El tiempo transcurrido entre las ubicaciones se puede obtener a través de las clases de Google, ya que las ubicaciones además de almacenar latitud, longitud y altitud, permiten el acceso al instante de tiempo en el que fueron tomadas. Una vez obtenidos la distancia y el tiempo transcurridos, se lleva a cabo la división de la distancia entre el tiempo, obteniendo la velocidad instantánea del corredor V_t .

Conocido el procedimiento de cálculo de la velocidad instantánea y la distancia recorrida durante el trayecto, se describe el método de cálculo del ritmo medio de la actividad. Para ello, definimos la variable interna T_p (tiempo de pausa).

Una de las funcionalidades implementadas en la pantalla de la actividad, es la posibilidad de pausar la carrera. Para calcular el ritmo medio, es importante tener en cuenta que, a pesar de que se dejen de actualizar las ubicaciones, el tiempo pausado se está teniendo en cuenta en T_F . El esquema de funcionamiento del tiempo de pausa se representa en la figura 11:

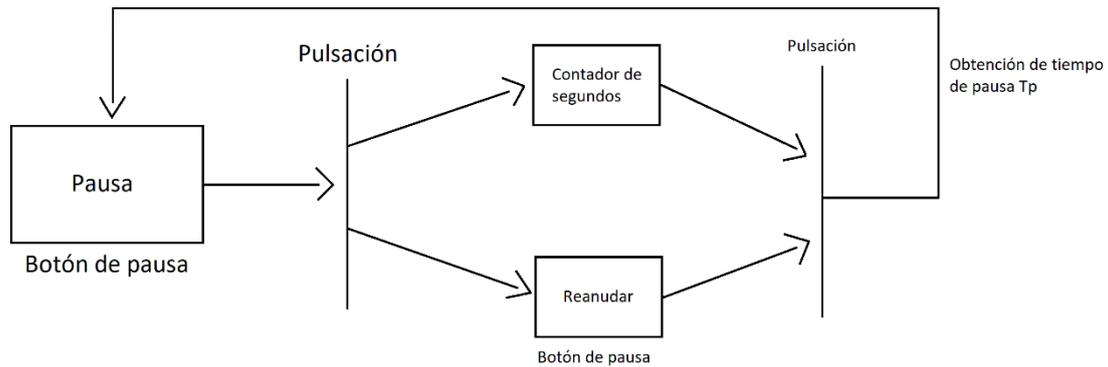


Figura 10. Esquema cálculo ritmo medio

Inicialmente, la actividad se encuentra actualizando las ubicaciones, en el momento en el que usuario decide pausarla, el texto del botón pasa a ser “Reanudar” y se comienza a contar el tiempo de pausa. Cuando el usuario decida reanudar la actividad, el conteo del tiempo de pausa parará obteniendo T_p de esa determinada pausa.

En caso de que el usuario realice posteriormente otra pausa, el valor inicial T_p será el valor final de la pausa anterior, teniendo en cuenta todas las pausas a la hora de calcular el ritmo medio.

En la computación del ritmo medio, se emplea el tiempo total de la actividad, mediante la comparación del instante de inicio de actividad T_0 , el tiempo de pausa T_p y el tiempo final de la actividad T_F , obtenido cuando se recorre la distancia total o se pulsa el botón de finalizar.

$$\text{Ritmo medio} = \frac{D_T}{(T_F - (T_0 + T_p))}$$

3.2.3. Detección de alertas

Una vez introducido el ritmo objetivo y calculado el ritmo a tiempo real del usuario, se procede a explicar el funcionamiento del sistema de alertas.

Para el envío de señales, como ya se ha comentado se han utilizado señales visuales y auditivas, que informen al usuario de su ritmo actual V_t con respecto al ritmo objetivo.

Tipos de señales auditivas

Se han implementado dos señales auditivas distintas. Una de ellas es la utilizada como señal de alerta al usuario, cuando éste corre fuera del intervalo objetivo, independientemente sea a un ritmo superior o inferior. La otra señal auditiva se emplea cuando el corredor finaliza la actividad. Esto puede darse de dos formas distintas, mediante la pulsación del botón de finalización de actividad o cuando la distancia total D_T es superior a la distancia recorrida objetivo.

Tipos de señales visuales

En el funcionamiento de las señales visuales se han implementado dos escenarios, que son el cambio de color de la pantalla al color rojo y al color amarillo.

La alerta visual de pantalla roja se implementa cuando el usuario corre a un ritmo inferior al objetivo, mientras que la pantalla amarilla se implementa cuando corre a un ritmo superior al objetivo. Los colores seleccionados son el amarillo y rojo debido a que son colores llamativos y comunes, de forma que serán fácil de asociar por el usuario, a un intervalo de ritmo concreto.

El funcionamiento de las señales se muestra en la figura 12:

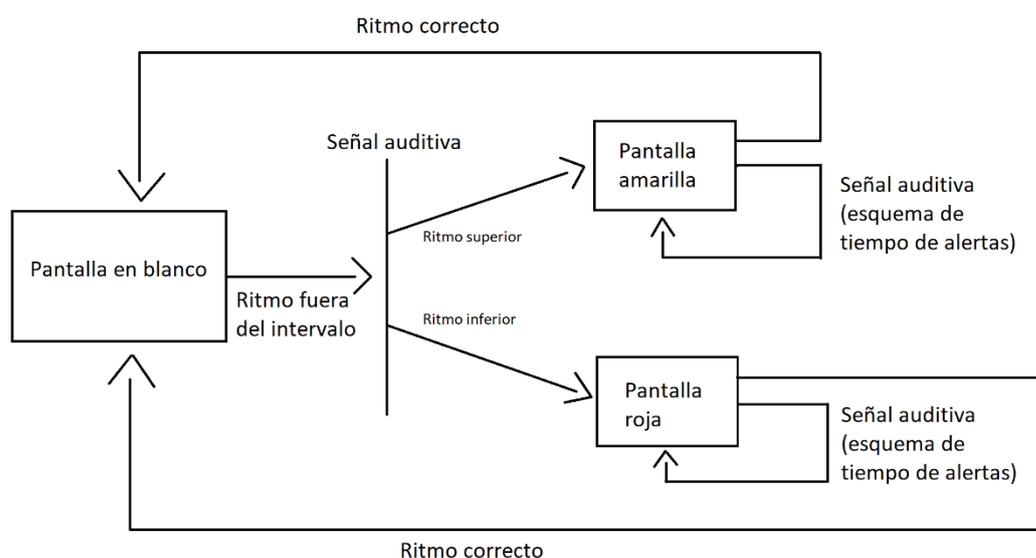


Figura 11. Esquema funcionamiento señales

Como se puede observar, cuando el corredor se sale del intervalo objetivo, se envía una señal auditiva, en función de si el ritmo es superior o inferior al objetivo, el fondo de pantalla se pondrá de color amarillo o rojo. Si el usuario continúa corriendo fuera del intervalo objetivo, la pantalla permanecerá en el color de alerta y se irán enviando señales auditivas siguiendo el esquema de la figura.

El esquema de funcionamiento del tiempo entre alertas se representa en la siguiente figura 6:

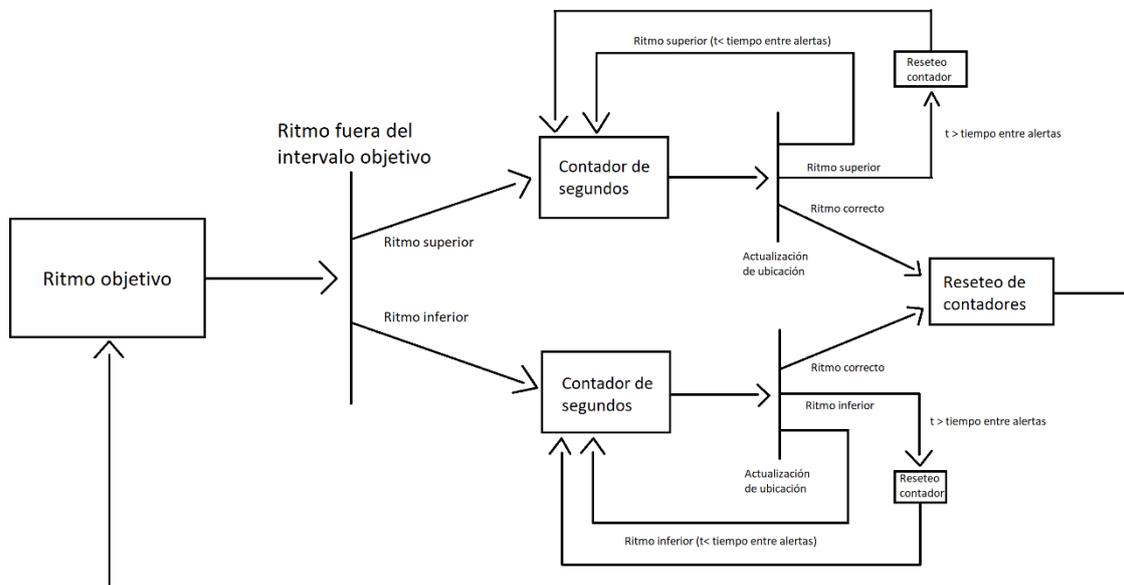


Figura 12. Esquema de funcionamiento del tiempo entre alertas

Como se puede observar en el momento en el que nos salimos del intervalo objetivo, se comienza a contar el tiempo que se permanece en dicho intervalo. Conforme se van realizando actualizaciones, se comprueba si el ritmo continúa fuera del intervalo objetivo. En ese caso se seguirá realizando la cuenta, hasta el momento en el que el tiempo sea mayor al tiempo entre alertas, momento en el que se resetea el contador y se envía una señal auditiva al usuario. Siempre que nos encontremos fuera del intervalo y se obtenga un ritmo correcto, se resetean los contadores de tiempo y se vuelve a la situación inicial, corriendo al ritmo objetivo.

3.2.4. Historial de actividades

Una vez finalizada la actividad ésta es almacenada de forma que seamos capaces de mostrar al usuario las actividades realizadas previamente.

En el historial de actividades se almacenan los siguientes datos:

- Ritmo
- Fecha
- Distancia recorrida
- Número de veces fuera del intervalo objetivo

En el historial se exponen los parámetros calculados previamente del ritmo V_m y la distancia recorrida V_t que junto con la fecha, servirán al usuario para constatar sus progresos deportivos.

El número de veces fuera del intervalo objetivo muestra al usuario las veces que ha rebasado el intervalo de tiempos tanto por el límite superior, como por el límite inferior. Este parámetro es interesante para el corredor, ya que también le ayuda a conocer si está realizando progresos, ya que una actividad realizada de forma óptima debería salirse del intervalo objetivo un escaso número de veces.

El historial de actividades se almacena en un fichero, dotado inicialmente del formato conveniente para almacenar los datos requeridos y que se puedan mostrar posteriormente de forma lógica al usuario. Se ha diseñado de forma que la última actividad que el usuario ha realizado es la primera en ser mostrada, pudiendo observar las siguientes actividades pulsando un botón implementado en esa pantalla.

3.3. DIAGRAMA DE NAVEGACIÓN DE LA APLICACIÓN

A continuación, se presenta el diagrama de navegación de la aplicación a través del cual se está representando la estructura general del sistema. De esta forma obtenemos una representación visual de los contenidos y de forma rápida podemos obtener una primera idea de las distintas pantallas que se han creado para el funcionamiento de la aplicación.

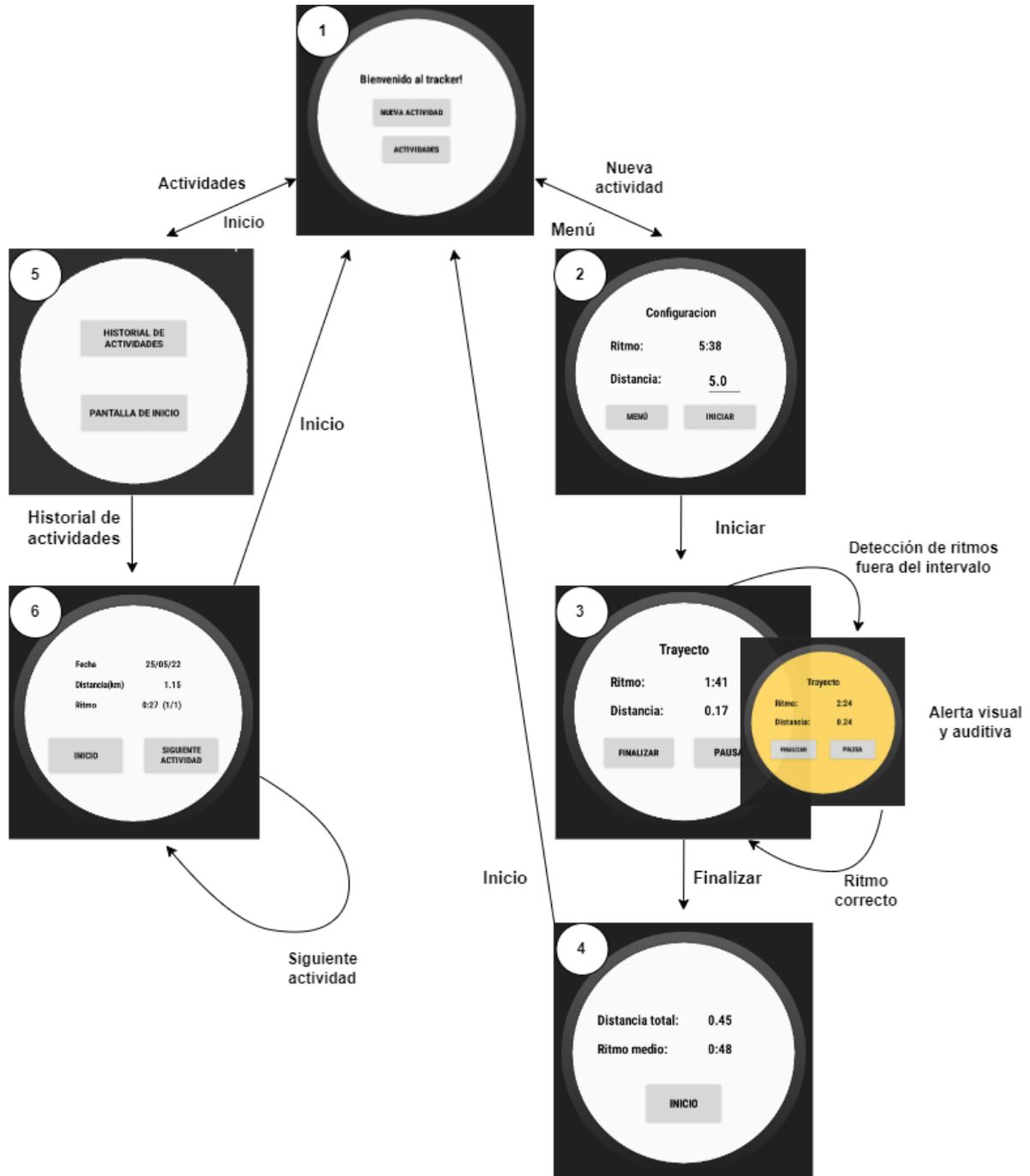


Figura 13. Diagrama de navegación

3.3.1. Descripción diagrama de navegación

Inicialmente, la aplicación se abre y nos muestra la pantalla de inicio (1), en la cual se nos presentan dos opciones, ir hacia la pantalla para comenzar la nueva actividad o dirigirnos a la pantalla en la cual se nos mostrarán las actividades que se han ido realizando previamente en la aplicación.

El funcionamiento de la aplicación a través del cual el usuario interactúa con la aplicación es mediante los botones dispuestos en la pantalla de la figura 14, que responden a las pulsaciones del usuario.

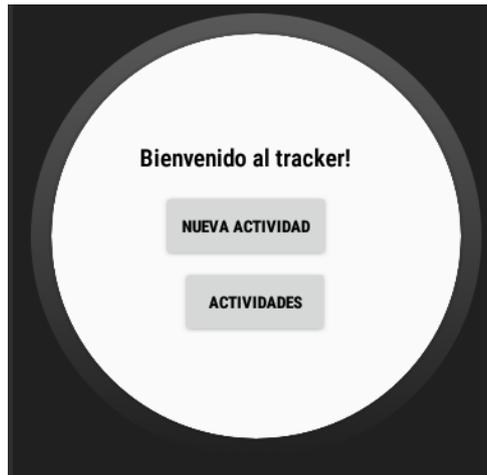


Figura 14. Menú de inicio

3.3.1.1. Inicio de Nueva Actividad

Si el usuario quiere realizar una nueva actividad, mediante la pulsación del botón "Nueva actividad" se dirigirá a la siguiente pantalla (2).

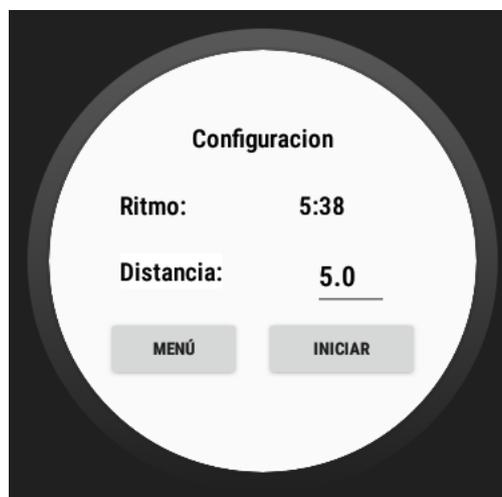


Figura 15. Inicio actividad

En esta pantalla (figura 15) se permite al usuario la configuración tanto del ritmo que quiere establecer como objetivo, como de la distancia que quiere recorrer al iniciar la actividad. Para la configuración de estos parámetros el usuario debe presionar los tiempos mostrados en pantalla, ya que esta es la forma más intuitiva al intentar modificar esos parámetros. En esta pantalla se da la posibilidad de dirigirse nuevamente a la pantalla de inicio, descrita previamente, mediante el botón con el texto “Menú” o de comenzar la actividad mediante el botón con el texto “Iniciar”.

En caso de que el usuario comience la nueva actividad se dirigirá a la siguiente pantalla (3), donde puede observar permanentemente la velocidad a la que se encuentra corriendo y la distancia que ha recorrido desde el inicio de la actividad.



Figura 16. Ritmo objetivo

En esta pantalla (figura 16) es en la cual se hacen más notorias las mejoras que se implementan en el proyecto. En ésta es en la cual se emiten las señales cuando el corredor se sale del ritmo establecido como objetivo.



Figura 17. Actividad con ritmo inferior



Figura 18. Actividad con ritmo superior

Como se puede observar en las figuras 17 y 18, cuando el ritmo es superior al establecido como objetivo, es decir, vas a mayor velocidad de la estipulada, la pantalla se pone de color amarillo y realiza una señal acústica, para que de forma rápida el corredor conozca que está yendo a un ritmo superior al establecido. En caso contrario, si el corredor va a un ritmo inferior del establecido, de forma paralela al caso anterior, una señal acústica avisará al corredor del ritmo incorrecto y además la pantalla se pondrá de color rojo.

En esta pantalla únicamente se permite dirigirse a la siguiente actividad, una vez finalizada la actividad. El otro botón únicamente producirá la parada de la aplicación, de forma que se deje de contar la distancia recorrida y el tiempo que transcurre durante el periodo de tiempo en el cual se encuentre pausada la aplicación.



Figura 19. Actividad principal

Como puede observarse en la figura 19, mediante la pulsación del botón pausar, cambia el texto del mismo para indicar al usuario que para retomar la actividad debe volver a presionar el botón, momento en el cual volverá a cambiar el texto a “Pausar”.

La finalización de la aplicación puede deberse a que el corredor presione el botón de finalizar, porque desee finalizar el trayecto previamente a haber cumplido el objetivo de kilómetros o porque la aplicación detecte que la distancia total recorrida es igual o superior a la distancia que se había marcado como objetivo, caso en el cual finaliza directamente la aplicación. Ambos casos dan paso a la pantalla representada en la figura 20.



Figura 20. Pantalla de ritmo medio

Una vez el recorrido se ha finalizado se muestra la pantalla en la cual se exponen al usuario la distancia total que ha recorrido y el ritmo medio que ha llevado durante todo el trayecto (figura 20). Cuando el usuario desee podrá volver a la pantalla inicial (1) presionando el botón de “Inicio”.

3.3.1.2. Historial de Actividades

En el caso de que el usuario quiera visualizar las actividades que ha realizado previamente, deberá pulsar el botón “Actividades” en la pantalla de Inicio, dirigiéndose a la pantalla (5) que le permite tanto volver a la pantalla inicial como ir a la pantalla del historial de actividades (figura 21).



Figura 21. Pantalla previa historial de actividades

En caso de que el usuario se dirija al historial de actividades mediante el botón “Historial de Actividades” visualizará la pantalla representada en la figura 22.



Figura 22. Historial de actividades

En esta pantalla (figura22) se muestra al usuario el historial de actividades. Inicialmente se observa la última actividad registrada en el dispositivo. De la última actividad registrada se muestra la fecha en la que realizó la actividad, la distancia total que recorrió y el ritmo medio. Además, se muestra colindante al ritmo la cantidad de veces que el corredor sobrepasó el intervalo de ritmo establecido, tanto por ir a ritmos superiores como por ir a ritmos inferiores.

En esta pantalla (figura 22) se permite ir observando progresivamente las actividades que se fueron realizando anteriormente mediante el botón de “Siguiete Actividad” o volver a la pantalla de inicio.

3.4. ESTUDIO DE USABILIDAD

En este apartado se va a realizar un estudio de usabilidad de la aplicación mediante una encuesta. Se pretenderá obtener información de distintos aspectos de la aplicación:

- **Usabilidad:** se pretenderá corroborar si la aplicación es intuitiva, es decir, si es fácil de utilizar incluso para usuario con escaso conocimiento en estas aplicaciones.
- **Implementación de señales de mejora:** se quiere recabar información de si los usuarios creen que mejoran la experiencia en el uso de actividades de control de ritmos y si creen que son las adecuadas o se podrían implementar otras.
- **Estética:** se quiere conocer la percepción estética de los usuarios de la aplicación que se ha desarrollado
- **Satisfacción general:** conocer la satisfacción del usuario a niveles generales con la aplicación.

El número total de usuarios que ha participado en la encuesta, es de 18. El número es bajo debido a la escasez de tiempo para la realización de la encuesta y la dificultad de encontrar personas con dispositivos que permitiesen la instalación de la aplicación.

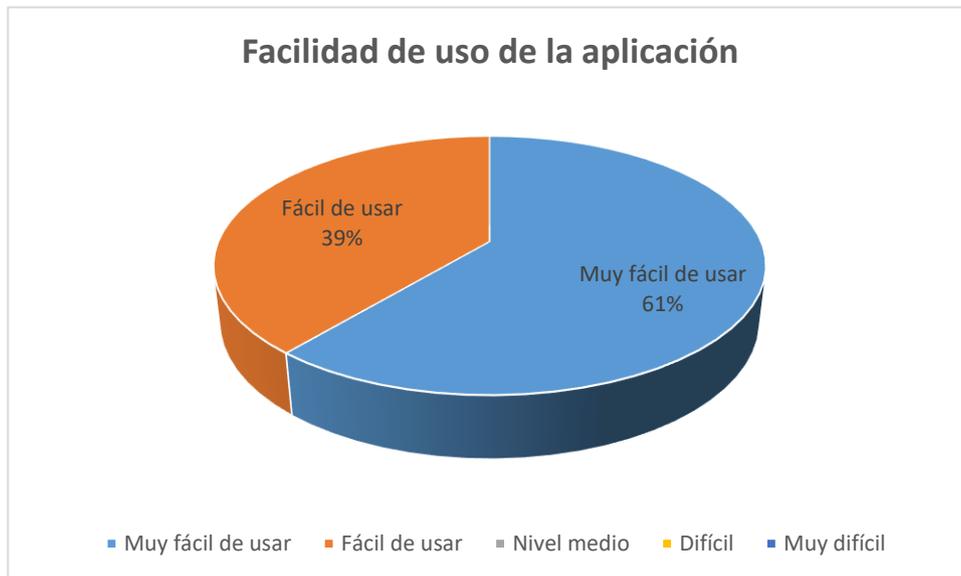


Figura 23. Facilidad de uso de la aplicación

En general, como se puede observar en la figura 23, los usuarios perciben que la aplicación es muy fácil de usar e intuitiva (61%). El 39% de los usuarios perciben que es fácil, mientras que ningún usuario percibe que tenga un nivel medio o mayor de dificultad.

Las respuestas cumplen con los objetivos que se habían establecido a la hora de desarrollar la aplicación. Ésta fue diseñada para que usuarios con escaso conocimiento pudiesen controlar la aplicación de forma sencilla.



Figura 24. Diseño de la interfaz

En cuanto al diseño de la aplicación, hay mayor diversidad de opiniones (figura 24). Un 39% creen que es muy acertado el diseño, un 44% creen que es un buen diseño mientras que un 17% creen que es correcto.

Teniendo en cuenta los resultados obtenidos, se observa margen de mejora en los aspectos estéticos. Se podría rediseñar la aplicación para intentar mejorar la percepción en este aspecto de los usuarios.

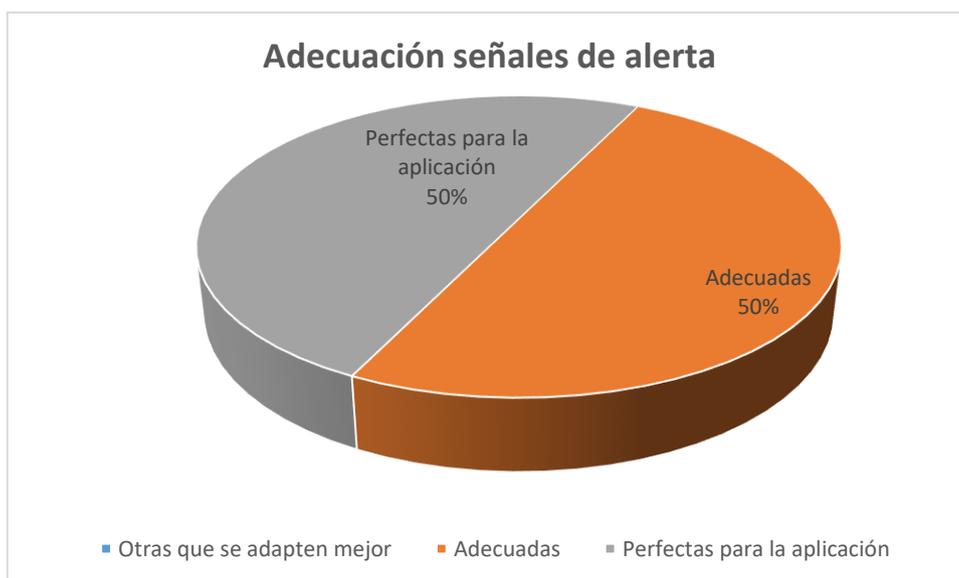
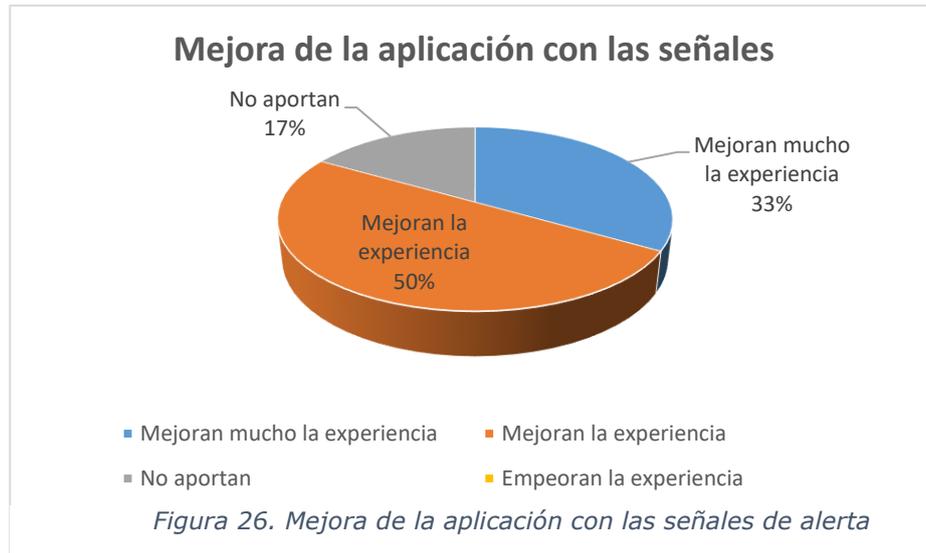


Figura 25. Adecuación señales de alerta

Otra de las cuestiones realizadas, fue la adecuación de las señales de alerta utilizadas en la aplicación (figura 25). Un 50% de los usuarios creen que eran perfectas para la

aplicación, mientras que un 50% creen que simplemente eran adecuadas. En principio observando estos resultados no se ve necesario la realización de cambios en las señales de alerta implementadas.

En



cuanto a la mejora de la aplicación con las señales (figura 26). Un 50% creen que mejora la experiencia, un 33% creen que mejora mucho la experiencia y un 17% cree que las señales no mejoran el uso de la aplicación.

En la figura 25, se observa como todos los usuarios creen que las señales son adecuadas, sin embargo, un 17% de los usuarios al utilizar la aplicación terminan opinando que no aportan nada. Esto es debido a la interacción que existe entre la aplicación, las señales y el usuario. Interacción que se comentará más adelante como posibles mejoras a la aplicación actual.



Figura 27. Satisfacción general con la aplicación

Como puede observarse en la figura 27, la aplicación tiene buena acogida entre los usuarios. Un 44% de los usuarios han respondido que en caracteres generales les encanta, a un 50% les gusta y a un 6% les es indiferente.

En general, a pesar de no haber podido realizar un número elevado de encuestas, los datos obtenidos son positivos. La elección de las señales e implementación parece correcta. Según los resultados se debería realizar ciertas modificaciones en la interfaz de la aplicación y en las interacciones entre la aplicación y el usuario, para mejorar la satisfacción del usuario durante el uso de la aplicación.

3.5. DEFINICIÓN DE ENSAYOS A REALIZAR

Inicialmente, se planteó una batería de ensayos teórica para comprobar el funcionamiento de la aplicación. Mediante estos ensayos se tratan de comprobar los aspectos básicos de la aplicación y las mejoras que se han implementado en este proyecto. Los ensayos a realizar son los siguientes:

-Funcionamiento básico: para comprobar el funcionamiento más básico de la aplicación, se correrá una distancia de un kilómetro con el dispositivo secundario de ayuda y el reloj. Se comprobará que la distancia total recorrida en el reloj es la misma que la distancia obtenida en el dispositivo de ayuda.

Lo segundo que se quiere comprobar es si realmente se ejecutan de forma correcta las mejoras de nuestra aplicación. Se tratará de verificar mediante las pruebas la correcta actividad del sistema de alerta por sonidos y del cambio de color del fondo de pantalla de la actividad.

Las pruebas a realizar serán las siguientes:

-Permanecer 1 minuto a una velocidad menor de la objetivo: debería escucharse el sonido de alerta la primera vez que entremos en el intervalo de velocidad baja y permanecer con el color de la pantalla en naranja, indicador de velocidad baja. Pasado un periodo de tiempo de en torno a 30 segundos debería escucharse de nuevo el pitido, para alertar al corredor de que permanece yendo a un ritmo incorrecto.

Mediante esta prueba se pretende comprobar el funcionamiento tanto del sistema de pitidos como del sistema visual cuando el corredor va a velocidades menores de la objetivo.

-Permanecer 1 minuto a una velocidad mayor de la objetivo: se verifican los mismos aspectos de la aplicación que en la prueba anterior pero en el límite de velocidades opuesto.

-Permanecer 1 minuto a la velocidad objetivo: en esta prueba se comprueba que en la velocidad objetivo el color de la pantalla permanece en blanco, de forma que no alerte al corredor y no se produzcan señales sonoras, indicando que estas yendo a una velocidad correcta.

En esta prueba el único pitido que se debería escuchar es el de finalización de actividad.

-Cambios en los límites de velocidad: en estas pruebas se tratará de verificar que el reloj responde de forma adecuada ante cambios en la velocidad con entradas y salidas en los intervalos de velocidad menor, mayor y velocidad objetivo. En estas pruebas se definirá un ritmo objetivo elevado de forma que se pueda ir a una velocidad considerablemente más rápida a la objetivo de forma fácil. Se realiza de esta forma para hacer cambios de ritmo bruscos y dar facilidad de detección a la aplicación.

Se realizará una prueba en la cual nos encontremos corriendo al ritmo objetivo y modifiquemos la velocidad para ir a un ritmo más lento del objetivo. La aplicación al notificar el cambio debería enviar una señal auditiva y cambiar el fondo de pantalla a un color amarillo. De forma análoga se debe realizar esta prueba para un ritmo más rápido del objetivo.

-Número de veces fuera del intervalo objetivo: se realizarán varias pruebas saliéndonos un número predeterminado de veces por ambos límites, apoyándonos en el dispositivo de ayuda, de forma que seamos conocedores de cuantas veces hemos pasado el límite inferior y superior de velocidad, pudiendo comprobar al final del trayecto los valores obtenidos.

-Tiempo de adaptación: se debe comenzar la actividad y verificar que efectivamente se produce una actualización de los datos de posición y se muestra por pantalla la velocidad del usuario y la distancia que recorre. Durante el tiempo de adaptación no se debería atender a los requisitos del ritmo objetivo, de forma que el dispositivo no debería enviar señales al usuario. Además, es importante comprobar que

posteriormente al tiempo de adaptación la aplicación se actualiza correctamente y ninguna de las variables que posteriormente se necesitarán para la señalización se ha visto modificada.

-Historial de actividades: se debe comprobar mediante la realización de varios trayectos que el historial de actividades muestra la información de forma coherente con las actividades hechas. Deben ser mostradas en el orden correcto en el que se han realizado y de forma secuencial, para permitir una buena visibilidad de las mismas.

-Finalización de la actividad: es importante comprobar que la actividad finaliza cuando se llega al límite de distancia establecido. Para ello, se establecerá una distancia de trayecto baja, de 500 metros para comprobar que la aplicación finaliza de forma automática al acumular esa distancia y envía la señal auditiva de finalización.

-Sistema de paradas: la última prueba a realizar consiste en la verificación del sistema de paradas. Para la realización de esta prueba será necesaria la utilización del dispositivo secundario de ayuda. Se recorrerá una distancia de un kilómetro, realizando pausas durante el trayecto. Es importante que la aplicación auxiliar tenga implementado este sistema de paradas, para pausar ambas aplicaciones al mismo tiempo. Se verificaría que el sistema de pausas está bien implementado, si al final del kilómetro, los datos mostrados de ritmo y distancia coinciden en ambos dispositivos.

Mediante el cumplimiento de estas pruebas se estaría garantizando la operatividad de todas las funcionalidades que la aplicación debía cumplir mediante las primeras versiones completas de la aplicación.

4. EXPERIMENTOS

En el capítulo anterior se definieron los ensayos teóricos que deberían realizarse para comprobar el funcionamiento íntegro de la aplicación. En este capítulo se describe la ejecución de dichos experimentos.

Los ensayos, se tuvieron que adaptar a las limitaciones del simulador y del reloj. Utilizando las propuestas teóricas como punto de partida, se modificaron para ajustarlas a los dispositivos de experimentación, comprobando las funcionalidades previstas.

4.1. EXPERIMENTOS EN EL EMULADOR

El emulador tiene la capacidad de reproducir un mapa, permitiendo la introducción manual de ubicaciones. Estas ubicaciones, de igual forma que en los dispositivos reales, permiten el acceso a la longitud, latitud y tiempo, variables que se necesitan para el cálculo de los ritmos durante la carrera.

Los primeros experimentos de funcionamiento se llevaron a cabo en el emulador. Mediante estos experimentos se comprobaron los funcionamientos básicos del programa, para posteriormente llevar a cabo la instalación en el dispositivo real garantizando el funcionamiento básico de la aplicación.

-Primera prueba: La primera prueba que se realizará es el funcionamiento de la toma de distancias, lo cual directamente verifica una correcta gestión de las ubicaciones que se van obteniendo. Además de ello se comprobará si el cálculo del ritmo a tiempo real que realiza la aplicación es el adecuado.

Para ello nos ubicaremos en una zona donde seamos conocedores de las distancias para poder comprobar que se están calculando bien, además de ello sabiendo cómo se realizan los cálculos del ritmo se podrá verificar si se calcula de forma correcta. Para ello se procederá a desplazarse **200 metros**, conociendo que el cálculo del ritmo se realiza con las últimas 5 ubicaciones y un periodo de 4 segundos entre toma de datos el ritmo que debería obtenerse es el siguiente:

$$\frac{16 \text{ segundos}}{200 \text{ metros}} = 0.08 \frac{\text{segundos}}{\text{metro}} * 1000 \frac{\text{metros}}{\text{kilómetro}} = 80 \frac{\text{segundos}}{\text{kilómetro}}$$

Se debería obtener un ritmo de 80 segundos/kilómetro que para pasarlo a un sistema más convencional se transforma de la siguiente manera:

$$80 \frac{\text{segundos}}{\text{kilómetro}} * \frac{1 \text{ minuto}}{60 \text{ segundos}} = 1.3333 \frac{\text{minutos}}{\text{kilómetro}}$$

Que obteniendo la parte entera y multiplicando el resto por 60 se obtiene un ritmo de:

$$1:20 \frac{\text{min}}{\text{km}}$$



Figura 28. Primera prueba

Como puede observarse en la figura 28 tanto la distancia recorrida como el ritmo se calculan de forma correcta.

-Segunda prueba: la segunda prueba consistirá en cerciorarse de que las mejoras, objeto del proyecto, funcionan de la forma esperada. Para ello se establecerá un ritmo de 4:30 y nos situaremos fuera del rango aceptable de ritmos, es decir fuera del intervalo 4:20 - 4:40, de forma que por debajo de 4:20 debería cambiar el color de la pantalla a amarillo por ir excesivamente rápido y por encima de 4:40 rojo, por ir excesivamente lento, además, el reloj debería pitar al producirse esos cambios. Para completar la segunda prueba, al inicio de la misma comprobaremos que se ejecuta de forma correcta el intervalo de adaptación, yendo a ritmos más bajos durante los primeros 30 segundos, de forma que el reloj no debería realizar pitidos ni cambiar de color.



Figura 29. Segunda prueba

Como puede observarse en la figura 29, inicialmente a pesar de ir a un ritmo más lento que el estimado la pantalla no emite señal visual, ya que se encuentra en el periodo de adaptación. En el momento que se detecta un ritmo más alto, la pantalla se pone de

color amarillo, caso de ritmo 3:39. Cuando va a un ritmo más bajo la pantalla se pone de color rojo, avisando al corredor que tiene que acelerar.

-Tercera prueba: en la tercera prueba se comprobará el funcionamiento del historial de actividades según se ha previsto. El historial de actividades se ha desarrollado de forma que debe mostrar la fecha en la que se ha realizado la sesión, la distancia total que ha recorrido y el ritmo medio de la actividad. Además, se comprobará que las actividades se guardan y muestran de forma secuencial, de forma que la última actividad sea la última mostrada y se puedan ver todas secuencialmente. Para comprobar que obtiene los datos correctamente se proceden a realizar dos simulaciones consecutivas.



Figura 30. Tercera prueba actividad 1

El primer experimento se realizó a día 09/06/2022, corriendo una distancia de 0.25 km (figura30). El segundo, se llevó a cabo de forma consecutiva, el propio día 09/06/2022, desplazándose una distancia de 0.78 km (figura 31).



Figura 31. Tercera prueba actividad 2

Como se puede comprobar, el historial de actividades se muestra de forma correcta, secuencial como se había previsto y mostrando los datos de la forma que se había diseñado.

-Cuarta prueba: en este experimento se tratará de comprobar una de las últimas implementaciones que se realizó, que es el conteo del número de veces que el corredor se sale del intervalo de tiempos establecido, diferenciando entre ritmos por encima y por debajo del objetivo. El número total de veces que se haya salido del intervalo por cada uno de los límites se muestra en el historial de actividades.

El funcionamiento esperado es que la aplicación no cuente cada vez que actualiza la posición si el corredor está dentro o fuera del intervalo, sino que, si el corredor permanece en uno de los intervalos erróneos de forma constante, aumente el contador según un intervalo predefinido en la programación. Por ello se tratará de hacer una actividad que se salga del intervalo por ir a una velocidad superior 4 veces y 3 veces por el intervalo inferior.



Figura 32. Cuarta prueba

Como puede observarse en la figura 32 la captura de datos ha sido correcta. El primer dato mostrado posterior al ritmo es el número de veces que se ha ido a un ritmo más rápido del ritmo objetivo, mientras que el segundo dato, muestra el número de veces que se ha ido a un ritmo más bajo.

A través de estos cuatro experimentos se comprueba el funcionamiento de la aplicación en los ámbitos más generales, obtención de ubicación, ritmos, funcionamiento de la interfaz y del historial de actividades.

4.2. EXPERIMENTOS EN EL RELOJ

Una vez realizadas las comprobaciones del código de la aplicación a través del emulador, se comprobó el funcionamiento de éste en dispositivos reales. Además, se comprobó que el diagrama de navegación y la interfaz se mostraban como se había configurado inicialmente. Para los experimentos funcionales, de gestión de ritmos de carrera, se utilizaron aplicaciones de apoyo, como Strava [37] y Adidas Running [38], permitiendo verificar la correcta obtención de datos durante el experimento.



Figura 33. Petición de permisos

Como puede observarse en la figura 33 el reloj de forma automática pide al usuario los permisos de ubicación la primera vez que abrimos la aplicación. Permiso obligatorio que debe aceptar el usuario para que se pueda acceder a la ubicación durante la actividad.

Pantallas Iniciales y configuración

La primera comprobación consistió en verificar que el sistema de navegación definido previamente, funcionaba correctamente y permitía al usuario desplazarse por las pantallas que se habían programado, y éstas interactuaban de forma correcta.



Figura 34. Pantalla de inicio

En la figura 34, se muestra la pantalla de Inicio de la aplicación, tal y como se había previsto en el diagrama de navegación. Nos permite ir a la configuración de la nueva actividad o acceder al historial de actividades.



Figura 36. Configuración de ritmo

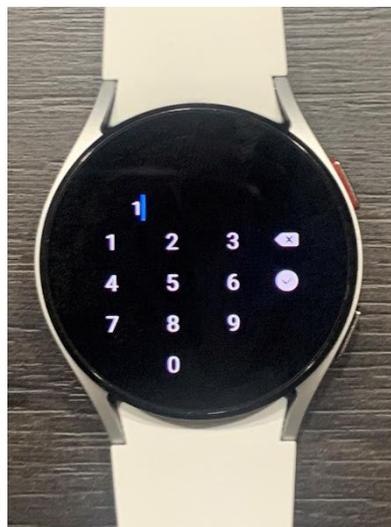


Figura 35. Configuración distancia

En la figura 35, se muestra la forma de configurar el ritmo de carrera, mientras que en la figura 36, se muestra la configuración de la distancia a recorrer durante la actividad.

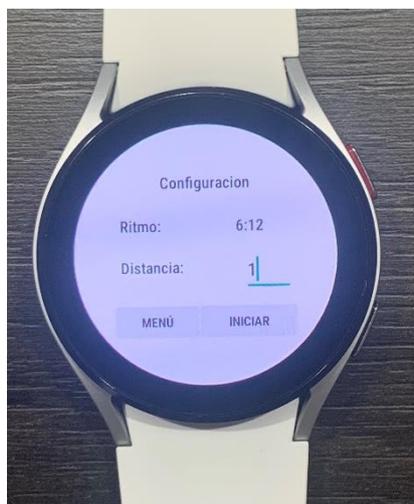


Figura 37. Pantalla de configuración

En la figura 37, se muestra la pantalla de configuración, una vez introducidos los datos de ritmo objetivo y distancia a recorrer. Cuando el usuario decida puede iniciar la actividad y comenzar el trayecto.

Actividad

En la actividad, como se había comentado, el fondo de pantalla cambia de color, según el ritmo que llevemos con respecto al ritmo objetivo. En este apartado se ha establecido un ritmo de 6:30, para ser capaces de superarlo fácilmente por ambos límites. La forma de visualización de los fondos se muestra en las figuras 38, 39 y 40:



Figura 40. Ritmo superior



Figura 39. Ritmo objetivo



Figura 38. Ritmo inferior

En la figura 40, se puede observar el color amarillo en la pantalla, indicando que el usuario está corriendo a una velocidad superior a la establecida. En la figura 39, el corredor está yendo al ritmo correcto, de forma que la pantalla permanece blanca, sin alertar al usuario. En caso de que el corredor vaya a una velocidad inferior, la pantalla se pone de color rojo, alertando al usuario, tal y como se muestra en la figura 38.

Historial de actividades

El historial de actividades debe mostrar la fecha en la que se ha realizado la actividad, la distancia recorrida y el ritmo que ha llevado el usuario. La disposición del historial de actividades es la mostrada en la figura 41:



Figura 41. Historial de actividades

Como puede observarse, se disponen los datos en forma vertical, para aprovechar el reducido espacio del reloj lo máximo posible. En la parte superior se muestra la fecha en la que se realizó la actividad, posteriormente se muestra la distancia recorrida y por último el ritmo de la carrera. Junto al ritmo de carrera, como información adicional se muestra el número de veces que el usuario ha sobrepasado los límites del intervalo de ritmo objetivo.

Pruebas de funcionamiento

Inicialmente, se realizó una prueba estableciendo una distancia a recorrer de 1 km para comprobar que obtiene la distancia y ritmo de forma correcta. Para ello se utilizó la aplicación de Strava en el móvil como aplicación de ayuda.



Figura 42. Actividad reloj



Figura 43. Actividad móvil

Como se puede observar en las figuras 42 y 43, la distancia recorrida en el reloj, una vez finalizada la actividad es de 1km, mientras que en Strava fue de 0.97km, es decir, hay un error del 3%. En cuando al ritmo medio, teniendo en cuenta el error en la distancia recorrida, Strava obtiene un ritmo de 12:30 min/km, mientras que en el reloj se obtiene un ritmo medio de 12:09. En esta primera prueba se constató que el reloj calcula la distancia de forma correcta, teniendo un error de en torno al 3% debido a la precisión del GPS y calcula el ritmo medio de forma correcta.

Se realizaron diversas pruebas para verificar el funcionamiento del sistema de alertas, tanto auditivas como sonoras. Para ello se permaneció durante 1 minuto a un ritmo superior al establecido, a un ritmo inferior y al ritmo objetivo. Posteriormente se realizaron cambios de ritmo, para comprobar que el sistema reaccionaba de forma correcta.

Las alertas visuales, es decir, los fondos de pantalla, se modificaban de forma correcta. Modificándose en función del ritmo del corredor, además si el corredor permanecía tanto al ritmo correcto como a un ritmo errónea, la pantalla permanecía en el color que se había programado. En el caso de las señales acústicas, cada vez que el corredor pasaba del ritmo objetivo a un ritmo inferior o superior, el reloj producía el sonido. Igualmente se comprobó que el dispositivo pitase cuando el corredor permanece fuera del intervalo objetivo durante un periodo de tiempo mayor de 30 segundos, obteniendo un resultado satisfactorio.

Una vez comprobado el sistema de funcionamiento básico, se realizaron pruebas para verificar el funcionamiento del sistema de paradas. Para ello, igual que en la anterior prueba mostrada, se utilizará una aplicación de apoyo, que disponga de sistema de paradas, para poder constatar el funcionamiento del sistema en la aplicación.

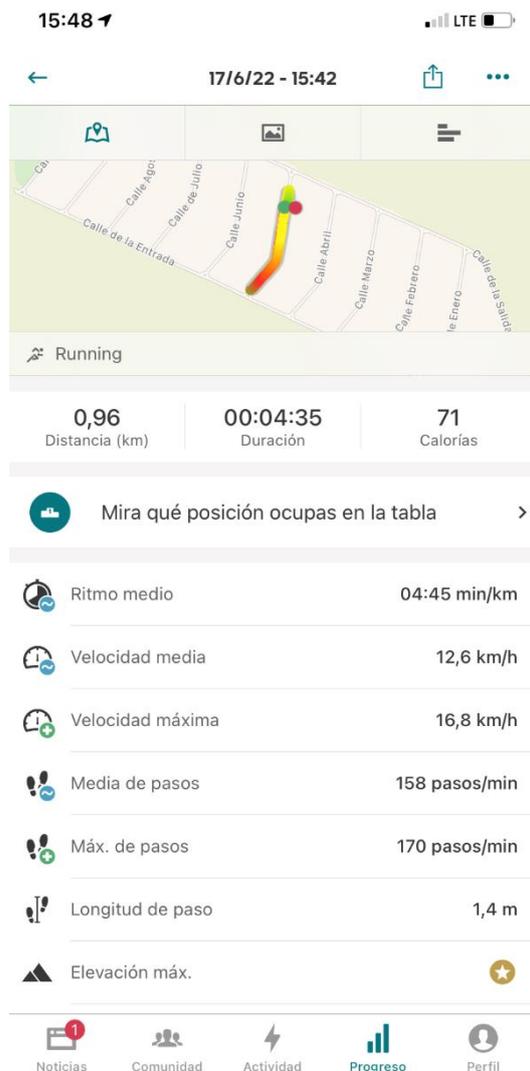


Figura 45. Actividad móvil con pausa 1

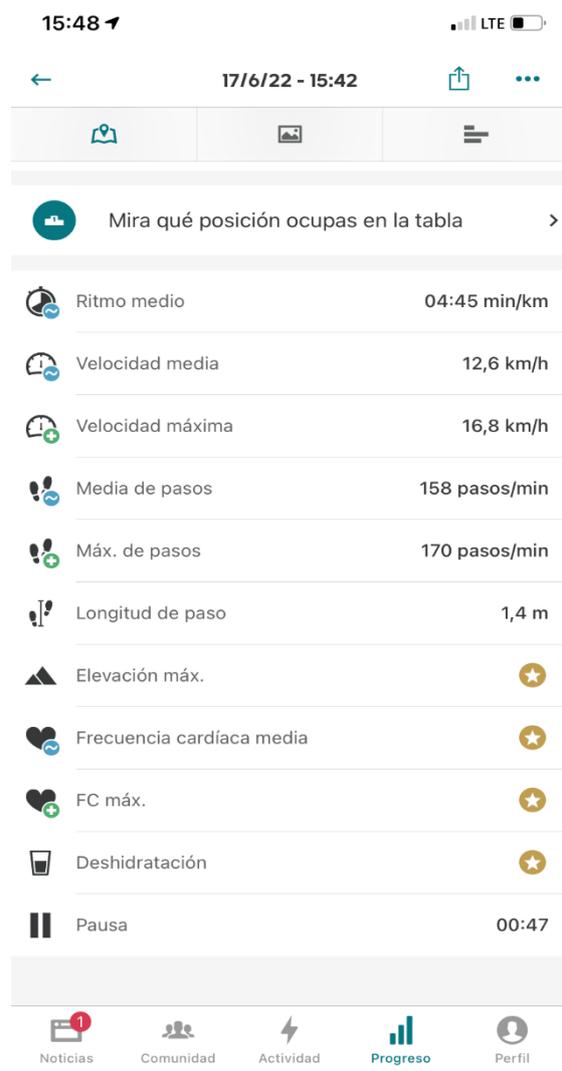


Figura 44. Actividad móvil con pausa 2

En las figuras 44 y 45 se observan las capturas de pantalla del trayecto realizado para la verificación del sistema de pausas. Los datos a tener en cuenta en estas figuras son, el ritmo medio durante el trayecto (figuras 44 y 45) y el tiempo que la actividad ha estado pausada, mostrado en la zona inferior de la figura 44.



Figura 46. Actividad con pausa reloj

Como puede observarse en la figura 46, la distancia recorrida en la actividad en el reloj es de 1.01km y el ritmo medio es de 4:32. Si observamos la aplicación de apoyo la distancia recorrida es de 0.96km y el ritmo durante el trayecto es de 4:45. Además, en la aplicación de apoyo se puede observar el tiempo que ha estado pausada la aplicación, en este caso 47 segundos, por lo tanto, si se ha obtenido un ritmo medio igual en ambas, es que el funcionamiento del sistema de pausa es correcto. Si el sistema de pausa no funcionase correctamente se obtendría un ritmo más lento en el reloj que en la aplicación de apoyo. La diferencia entre los ritmos de ambas, proviene como en el caso anterior de la diferencia en la distancia recorrida, ya que el reloj cuenta mayor distancia que el móvil y por lo tanto el ritmo medio es mayor.

5. GESTIÓN DEL PROYECTO

En este apartado se comentará la metodología que se ha llevado a la hora de realizar el trabajo de fin de grado, comentando las fases en las que se ha ido dividiendo el proyecto. Además de ello se expondrán las horas dedicadas a cada una de las actividades que se han llevado a cabo durante el proyecto.

5.1. ORGANIZACIÓN DEL PROYECTO

El proyecto comenzó a ser desarrollado en noviembre, momento en el cuál se comenzó a plantear como debía enfocarse el proyecto. Inicialmente debido a mi estancia en Génova durante el primer cuatrimestre, se dividió en dos partes, diferenciando la metodología durante el primer y segundo cuatrimestre.

En el primer cuatrimestre se tenía previsto un trabajo semanal de unas 15 horas, teniendo reuniones vía Google-Meet con el tutor de la asignatura los viernes, durante media hora, donde se comentarían los objetivos a cumplir durante esa semana, para tener unos objetivos a corto plazo más fáciles de cumplir y ser capaces de llevar un seguimiento del trabajo semana a semana.

Para el segundo cuatrimestre se estableció una metodología similar a la del primer cuatrimestre, se aumentaron las horas de trabajo semanal, debido a el acercamiento de la fecha de entrega, a 20 horas semanales, con el seguimiento del trabajo semanal a través de reuniones de entre 30 minutos y una hora con el tutor, donde de forma equivalente al primer cuatrimestre, se comentaban los avances durante la semana, las dificultades que se habían encontrado y se realizaban propuestas para resolverlas y se debían definir los objetivos a cumplir durante esa semana.

5.2. FASES DEL PROYECTO

El proyecto se dividió en diferentes fases, para su seguimiento e ir cumpliendo objetivos del mismo:

-Elección de dispositivo: el primer objetivo a cumplir fue seleccionar el reloj inteligente, intentando que cumplieres de forma conveniente con los requisitos que se habían establecido. Este fue el primer paso a dar, debido a que derivado de su elección se procedería a elegir el entorno y lenguaje en el que se tenía que desarrollar la aplicación.

-Aprendizaje e instalación Android Studio: inicialmente me instruí en básicos de Java, lenguaje en el que se iba a desarrollar la aplicación, intentando enfocar mi aprendizaje de la forma más eficiente para desarrollar el proyecto. Es decir, enfocándome en adquirir los conocimientos necesarios para desarrollar la aplicación objetivo.

-Programación: una vez obtenidos los conocimientos básicos en Java y Android Studio, se comenzó el desarrollo de la aplicación, fase a la cual se dedicó la mayor parte del tiempo del proyecto.

-Instalación y experimentos: una vez obtenida una aplicación funcional en los emuladores de Android Studio, se llevó a cabo la instalación de la aplicación en el reloj inteligente y el desarrollo de experimentos en el emulador y en el dispositivo real en post de comprobar el funcionamiento de la aplicación que se había estado desarrollando.

El total de horas dedicado fue de 305.5, distribuidas de la siguiente forma:

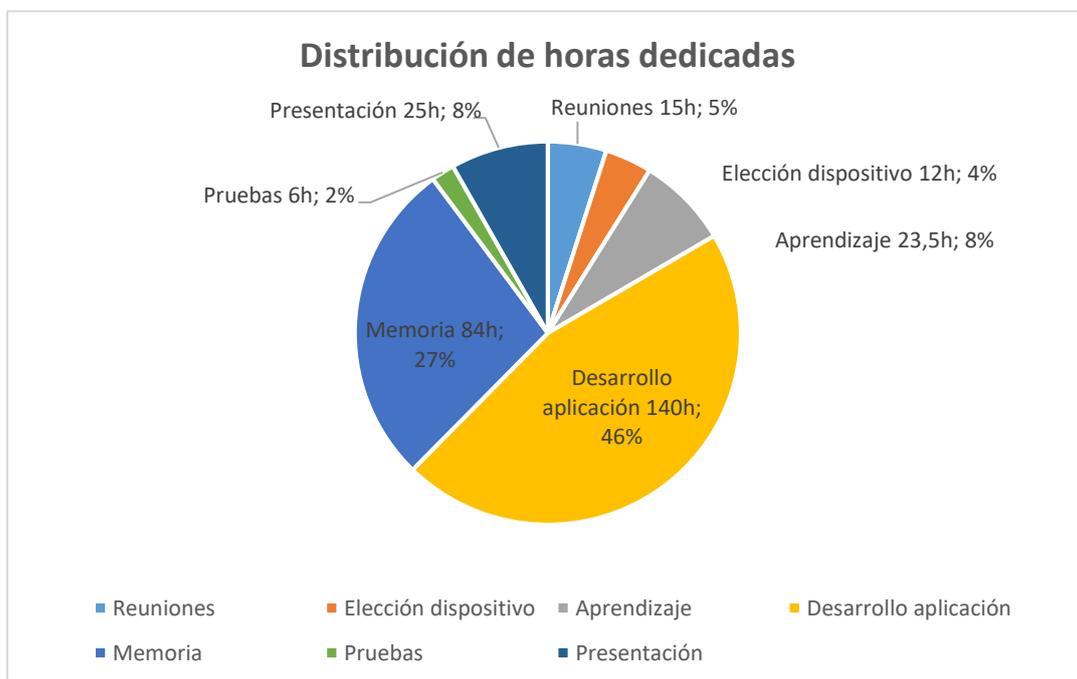


Figura 47. Distribución horas proyecto

6. CONCLUSIONES Y TRABAJO FUTURO

En este apartado se presentan las conclusiones extraídas durante la realización del trabajo y el conocimiento que se ha adquirido durante el desarrollo del proyecto. Finalmente se expondrán líneas de trabajo futuras que puedan mejorar y continuar con el desarrollo de este TFG utilizando éste como punto de partida.

6.1. CONCLUSIONES DEL PROYECTO

Durante el desarrollo del trabajo se ha conseguido crear una aplicación para dispositivos móviles, en este caso para relojes inteligentes que es capaz de realizar el seguimiento de la ubicación del usuario de forma precisa.

Inicialmente a través de las pruebas mediante el dispositivo de emulación se pudo ir haciendo un seguimiento de los progresos en la aplicación, hasta el punto en el cual se había desarrollado una aplicación lo suficientemente funcional para ser instalada en dispositivos reales. Los problemas surgieron debido al distinto funcionamiento entre la aplicación en dispositivos reales y los emuladores que generaron grandes problemas en la actualización de datos y obtención de ritmos en el dispositivo real. A pesar de ello, finalmente se consiguió obtener una aplicación funcional en el reloj que cumpliera con los requisitos de la aplicación que se había definido inicialmente.

6.2. CONOCIMIENTOS ADQUIRIDOS

Durante la realización del trabajo se adquirieron conocimientos técnicos y personales tanto durante las horas de trabajo en el desarrollo del código de la aplicación como en la escritura de la memoria.

6.2.1. Conocimientos Técnicos

1. Conocimientos básicos de lenguaje Java para su utilización en el código, como el uso de funciones, creación de clases y uso de clases externas predefinidas de Google, uso de atributos de clases.
2. Se aprendió a utilizar Android Studio, entorno en el cual se desarrolló la aplicación. Se aprendieron conceptos básicos como el uso de los bloques funcionales suministrados por la aplicación y se aprendió a modificar atributos de los bloques predefinidos para su adaptación al proyecto.
3. Se aprendió a utilizar Android Studio tanto para el desarrollo de proyectos orientados a aplicaciones móviles como a aplicaciones para relojes inteligentes.

4. Utilización de emuladores para realizar pruebas. Se aprendieron las funciones básicas del emulador, capaz de realizar los controles básicos de la aplicación y funciones más especializadas que permiten incluir datos específicos como la ubicación.
5. Instalación de aplicaciones mediante software externos como bugjaeger que permiten la conexión entre móvil y relojes inteligentes de forma que se pueden instalar las aplicaciones desarrolladas a través del APK obtenido en el entorno de desarrollo.
6. Capacidad de llevar un registro del tiempo empleado en las actividades para no incurrir en pérdidas de tiempo innecesarias o emplear excesivo tiempo en determinadas partes del proyecto.
7. Estructuración de proyectos de tamaño medio, así como desarrollo de tablas de contenidos que faciliten la visualización del mismo.
8. Redacción de documentos con carácter formal y exposición de las partes clave del mismo.

6.2.2. Conocimientos personales

1. Se ha aprendido a focalizar temas desde distintos puntos de vista para intentar desarrollar el proyecto de la forma más eficiente posible, de esta forma se han adquirido conocimientos de análisis y capacidad de planteamiento de los problemas que iban surgiendo durante el desarrollo del proyecto.
2. Capacidad de realizar un trabajo guiado y de comunicación con el tutor para avanzar y enfocar el proyecto de la forma más conveniente.
3. Valoración de imprevistos y gestión de los mismos y capacidad de modificación de las tareas y prioridades del proyecto.

6.3. DIFICULTADES DURANTE LA REALIZACIÓN DEL TRABAJO

Conforme el proyecto iba progresando fueron surgiendo problemas, los cuales se describen a continuación.

La escasez de conocimientos iniciales en Java, a pesar de que desde el primer momento traté de orientar mi aprendizaje para la realización del proyecto hizo que tuviese que emplear una gran cantidad de tiempo inicialmente para desarrollar y acomodarme a los métodos de trabajo del nuevo lenguaje.

Hubo problemas debido a la baja utilización de Android Studio para el desarrollo de aplicaciones para relojes inteligentes, ya que a pesar de que Android Studio está diseñada para el desarrollo tanto de aplicaciones móviles como de reloj, la gran parte de información para aplicaciones giraba en torno a aplicaciones móviles, por lo que se tuvo que hacer un aprendizaje más autodidacto. También se presentaron contratiempos relacionados con la adaptación de funciones sencillas como los visores de texto de Android Studio para los Smart-watch, debido a que no estaban correctamente adaptados ni optimizados para su visualización y manejo sencillo en este tipo de dispositivos.

Se tuvieron dificultades en la instalación de la aplicación debido a las bajas posibilidades que había y posteriormente, surgieron una gran cantidad de problemas de funcionamiento en la aplicación instalada en el reloj, que ralentizaron el proceso de desarrollo ya que en el dispositivo real no se tiene la capacidad de ver en qué punto está fallando el programa, ni cuál es el fallo. Además, la realización de cambios implica tener que reinstalar la aplicación, por lo que los fallos y cambios correctores que se realizaban en el reloj implicaban un tiempo de trabajo mucho mayor que los cambios en el emulador.

A pesar de estas incidencias se trató de reconducir en todo momento el proyecto de forma que se pudiese ir avanzando y acercando el proyecto lo máximo posible a los objetivos establecidos, obteniendo los mejores resultados posibles dentro del plazo de tiempo estipulado.

6.4. LÍNEAS DE TRABAJO FUTURAS

A continuación, se exponen las vías en las cuales se podría mejorar el proyecto actual.

Existe una interacción entre el dispositivo y la aplicación que provoca el mal funcionamiento de la aplicación durante la actividad deportiva. Si dejamos de interactuar con el dispositivo durante un periodo de tiempo elevado, el dispositivo vuelve de forma automática a su pantalla de inicio. Debido a esta interacción, la aplicación deja de tomar datos de ubicación, de forma que el ritmo instantáneo se ve modificado, ya que al volver a abrir la aplicación se da un salto en la ubicación, obteniendo un ritmo muy elevado. Se debería investigar esta interacción y tratar de ajustar que la aplicación pueda funcionar en segundo plano, o en su defecto, que no se sale de forma automática.

Se pueden realizar mejoras en el sistema de señales, mediante una diferenciación entre la señal auditiva de alerta por ritmo inferior y superior al objetivo. Mediante esta nueva señal, aumentaría mucho la información que se le da al corredor, ya que, sin mirar al reloj, sabría si está corriendo a un ritmo superior o inferior.

Finalmente, se propone realizar un nuevo estudio con mayor profundidad de los dispositivos, ya que el que se escogió, a pesar de en las especificaciones disponer de GPS, necesitaba de la conexión del móvil para funcionar. Se debería encontrar un dispositivo que además de cumplir con los requisitos que se han establecido en el trabajo, tenga conexión GPS autónoma.

7. BIBLIOGRAFÍA

- [1] Obtención de información respectiva al desarrollo de aplicaciones y clases de Google. Anónimo. 2012. *Android developers*. Disponible en: <https://developer.android.com/>
- [2] Pérez, E. (Productor). 2017. *Curso Android desde cero con Android Studio*. <https://www.youtube.com/watch?v=tyx05coXixw&list=PLyvsggKtwbLX06iMtXnRGX5lyjiiMaT2y>
- [3] García, J. Xataka, Samsung Galaxy Watch 4. Disponible en: <https://www.xataka.com/analisis/samsung-galaxy-watch-4-analisis-caracteristicas-precio-especificaciones>
- [4] Pérez, E. (Productor). 2017. *Curso Java desde cero* [YouTube]. Disponible en: <https://www.youtube.com/watch?v=L1oMLsiMusQ&list=PLyvsggKtwbLX9LrDn11-K6QtYo7m0yXWB>
- [5] Motorola Mobility LLC. 2022. *Motorola moto 360*. Disponible en: <https://www.motorola.es/new-moto-360/p>
- [6] Anónimo. 2016. *Curso programación Java desde cero*. Disponible en: <https://javadesdecero.es/>
- [7] Anónimo. 2020. *Wear OS*. Disponible en: https://es.wikipedia.org/wiki/Wear_OS
- [8] Bunton, C. 2022. *Sistema operativo Tizen*. Disponible en: <https://www.pocket-lint.com/es-es/smartphones/noticias/samsung/127527-que-es-tizen-y-en-que-dispositivos-aparecera>
- [9] Anónimo. 2022. *Android*. Disponible en: <https://es.wikipedia.org/wiki/Android>
- [10] Martín, I. 2021. *Mejores Smart-watch para instalar aplicaciones*. Disponible en: <https://topesdegama.com/listas/wearables/mejores-smartwatch-instalar-aplicaciones>
- [11] Penalva, J. 2016. *Moto 360 sport*. Disponible en: <https://www.xataka.com/analisis/moto-360-sport-analisis>
- [12] Pastor, J. 2020. *Características Oppo watch*. Disponible en: <https://www.xataka.com/relojes-inteligentes/oppo-watch-caracteristicas-precio-ficha-tecnica-1>

- [13] Martí.A. 2019. *Características Samsung Galaxy Watch Active 2*. Disponible en: <https://www.xataka.com/relojes-inteligentes/samsung-galaxy-watch-active-2-analisis-caracteristicas-precio-especificaciones>
- [14] React Native. 2022. *Reactnative*. Disponible en: <https://reactnative.dev/>
- [15] Qt Creator. 2022. *QtCreator*. Disponible en: <https://www.yeeply.com/blog/entornos-programacion-desarrollar-apps-android/>
- [16] Anónimo. 2022. *Android Studio*. Disponible en: https://es.wikipedia.org/wiki/Android_Studio
- [17] Yeeply. 2022. *Los 6 mejores entornos para crear aplicaciones Android*. Disponible en: <https://www.yeeply.com/blog/entornos-programacion-desarrollar-apps-android/>
- [18] PluginsXBMC. 2021. *Bugjaeger, instalación de APK's*. Disponible en: <https://www.pluginsxbmc.com/2021/01/Bugjaeger-descarga.html>
- [19] Samsung. 2022. *Galaxy Watch4 Bluetooth*. Disponible en: <https://www.samsung.com/es/watches/galaxy-watch/galaxy-watch4-black-bt-sm-r870nzkaphe/>
- [20] García, F (productor). 2015. *Música y sonidos en Android*. Disponible en: <https://www.youtube.com/watch?v=eVn6rZHu0iw>
- [21] Videvo. 2015. *Efectos de sonido*. Disponible en: <https://www.videvo.net/es/efectos-de-sonido/>
- [22] Anónimo. 2017. *Sonidos MP3 Gratis*. Disponible en: <http://www.sonidosmp3gratis.com/>
- [23] García, F (productor). 2015. *Manejo ficheros Android*. Disponible en: <https://www.youtube.com/watch?v=lbBosCrFE7Y>
- [24] Android desde cero (productor). 2019. *Tutorial Android Studio guardar y leer fichero*. Disponible en: <https://www.youtube.com/watch?v=-4SztKGpTBO>
- [25] Campos, M. 2016. *Generar APK con Android Studio*. Disponible en: <https://openwebinars.net/blog/generar-apk-android-studio/>
- [26] Sluiter, S. 2021. *Android Studio tutorial-Build a GPS App*. Disponible en: <https://www.youtube.com/watch?v=xUcYfbtfsI&pp=ugMICgJlcxABGAE%3D>
- [27] Sgoliver. 2016. *Localización Geográfica en Android*. Disponible en: <https://www.sgoliver.net/blog/localizacion-geografica-en-android-1/>

- [28] Linares, I. 2021. *Funcionamiento conexión GPS*. Disponible en: <https://www.xatakamovil.com/conectividad/asi-como-gps-tu-movil-consigue-localizarte-unos-pocos-metros-error-1>
- [29] Souza, I. 2019. *XML: ¿qué es y para que sirve?* Disponible en: <https://rockcontent.com/es/blog/que-es-xml/>
- [30] Java. 2022. *Java*. Disponible en: https://www.java.com/es/download/help/whatis_java.html
- [31] Murada, Y. 2020. *Que es el gradle: la herramienta para ser más productivo desarrollando*. Disponible en: <https://openwebinars.net/blog/que-es-gradle/>
- [32] Contenidos. 2020. *Google Play Store*. Disponible en: <https://neoattack.com/neowiki/google-play-store/>
- [33] Samsung Electronics Co., Ltd. 2022. *Galaxy wearable*. Disponible en: <https://play.google.com/store/apps/details?id=com.samsung.android.app.watc hmanager&hl=es&gl=US#:~:text=La%20aplicaci%C3%B3n%20Galaxy%20Weara ble%20conecta,haya%20instalado%20mediante%20Galaxy%20Apps.&text=%2 D%20Ajustes%20y%20tipo%20de%20notificaciones%2C%20etc.>
- [34] Strava, Inc. 2022. *Strava application*. Disponible en: <https://www.strava.com/features?hl=es-419>
- [35] Runtastic. 2022. *Adidas Runtastic*. Disponible en: <https://www.runtastic.com/es/>