



Universidad de Oviedo
Universidá d'Uviéu
University of Oviedo



Escuela de
Ingeniería
Informática
Universidad de Oviedo



Trabajo de
Desarrollo

TRAINING PARTNER: APLICACIÓN PARA MONITORIZAR TU SALUD Y TUS ENTRENAMIENTOS

TRABAJO DE FIN DE GRADO

AUTOR

Álvaro Aguilar-Galindo Marrodán

DIRECTOR

Miguel Sánchez Santillán

Julio 2023

Esta plantilla es copyright (C) 2019 JOSÉ MANUEL REDONDO LÓPEZ.

Teaching Innovation Project: PINN-19-A-029 (University of Oviedo) [1]

This work has been published in [2]

See the disclaimer document to know about how to handle licensing this document.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

El autor de esta plantilla declara expresamente no requerir que cada trabajo de fin de estudios lleve la misma licencia, por lo que no es necesario incluir el anexo de la licencia de la plantilla, salvo que quieras, por decisión personal, licenciar tu trabajo de la misma forma. Si, en cambio, lo que has hecho es un documento – plantilla a partir de este, debes respetar las condiciones de la GNU Free Documentation License, al considerarse en este caso un trabajo derivado de este documento.

Este documento ha sido creado basándose en la plantilla elaborada por JOSÉ MANUEL REDONDO LÓPEZ. [1] [2]



Declaración Responsable

El alumno: Álvaro Aguilar-Galindo Marrodán

Con DNI:

Y UO:

DECLARA

Que esta obra es completamente original y se han citado debidamente las fuentes utilizadas durante la realización de esta.

Y para que conste, lo firma en Oviedo, a 25 de 06 de 2023

Firmado:

Álvaro Aguilar-Galindo Marrodán



ÍNDICE DE CONTENIDO

¿Qué es este Trabajo?.....	25
Resumen.....	25
Palabras clave.....	25
Abstract	25
Keywords.....	26
Capítulo 1: Planificación del Sistema de Información.....	28
1.1. PSI 1: Inicio del Plan de Sistemas de Información.....	29
1.1.1. PSI 1.1: Análisis de la Necesidad del PSI.....	29
1.1.2. PSI 1.2: Identificación del Alcance del PSI	29
1.2. PSI 2: Definición y organización del psi	30
1.2.1. PSI 2.1: Especificación del Ámbito y Alcance	30
1.2.2. PSI 2.2: Organización del PSI	33
1.3. PSI 3: Estudio de la Información Relevante.....	34
1.3.1. PSI 3.1: Selección y Análisis de Antecedentes.....	34
1.3.1.1. Conceptos teóricos.....	34
1.3.1.1.1 Caloría.....	34
1.3.1.1.2 Acelerómetro.....	34
1.3.1.1.4 Localización y coordenadas	35
1.3.1.1.5 Estilo o capa de un mapa	36
1.3.1.2. Sistemas similares	36
1.3.1.2.1 Garmin connect.....	36
1.3.1.2.2 Strava.....	37
1.3.1.2.3 Polar flow	38
1.3.1.3. Comparación entre sistemas similares.....	38
1.3.1.4. Estudio de APIs relevantes	39
1.3.1.4.1 Google Maps	39
1.3.1.4.2 MapBox	39
Capítulo 2: Definición de la arquitectura tecnológica.....	42
2.1. PSI 7.1: Identificación de las Necesidades de Infraestructura Tecnológica	43
2.1.1. Aplicación móvil	43
2.1.1.1. Java con Android Studio	43
2.1.1.2. IOS Nativo con Swift	43
2.1.1.3. C# con Xamarin.....	44



2.1.2.	Base de datos	45
2.1.2.1.	Base de datos relacional	45
2.1.2.1.1.	SQLite	45
2.1.2.1.2.	MySQL	45
2.1.2.2.	Bases de datos documentales	46
2.1.2.2.1.	MongoDB	46
2.1.2.2.2.	Firebase	46
2.2.	PSI 7.2: Selección de la Arquitectura Tecnológica	47
Capítulo 3: Planificación del Sistema de Información		50
3.1.	Planificación del proyecto	51
3.1.1.	Identificación de Interesados	51
3.1.2.	OBS Y PBS	51
3.1.1.1	OBS	51
3.1.1.2	PBS.....	51
3.1.1.2.1	PBS estructura general	51
3.1.1.2.2	PBS Planificación del sistema	52
3.1.1.2.3	PBS Análisis del sistema de información	52
3.1.1.2.4	PBS Diseño del sistema.	53
3.1.1.2.5	PBS Desarrollo de software y pruebas	53
3.1.1.2.6	PBS documentación.....	54
3.1.1.2.7	Cierre del proyecto.....	54
3.1.3.	WBS y planificación inicial	55
3.1.1.3	WBS	55
3.1.1.3.1	Estructura general	55
3.1.1.3.2	Planificación inicial del sistema	55
3.1.1.3.3	Análisis del sistema de información	56
3.1.1.3.4	Diseño del sistema	56
3.1.1.3.5	Implementación del software y las pruebas	57
3.1.1.3.6	Documentación	58
3.1.1.3.7	Cierre del proyecto.....	59
3.1.1.4	Planificación inicial	59
3.1.1.4.1	Fase de planificación del sistema.	60
3.1.1.4.2	Fase de análisis.....	60
3.1.1.4.3	Fase de diseño	61
3.1.1.4.4	Fase de desarrollo	62
3.1.1.4.5	Fase de documentación	63



3.1.1.4.6	Fase de cierre del proyecto	63
3.1.1.5	Diagrama de Gantt	64
3.1.1.5.1	General	64
3.1.1.5.2	Fase de planificación	64
3.1.1.5.3	Fase de análisis	64
3.1.1.5.4	Fase de diseño	64
3.1.1.5.5	Fase de desarrollo	65
3.1.1.5.6	Fase de documentación	65
3.1.1.5.7	Fase de cierre de proyecto	65
3.1.4.	Riesgos	66
3.1.1.6	Plan de gestión de riesgos	66
3.1.1.7	Identificación de riesgos	66
3.1.5.	Presupuesto inicial	67
3.1.1.8	Presupuesto de costes	68
3.1.1.8.1	Partida 1: Planificación	69
3.1.1.8.2	Partida 2: Análisis	70
3.1.1.8.3	Partida 3: Diseño	71
3.1.1.8.4	Partida 4: Implementación	72
3.1.1.8.5	Partida 5: Documentación	74
3.1.1.8.6	Partida 6: Cierre de proyecto	75
3.1.1.8.7	Partida 7: Otros gastos	76
3.1.1.8.8	Resumen del presupuesto de costes	76
3.1.1.9	Presupuesto de Cliente	78
3.2.	Ejecución del proyecto	79
3.2.1.	Bitácora de incidencias	79
3.2.2.	Riesgos	80
3.3.	Cierre del proyecto	83
3.3.1.	Planificación final	83
3.3.2.	Presupuesto Final de Costes	86
3.3.3.	Comparativa de planificación y presupuesto inicial y final	87
3.3.4.	Informe de lecciones aprendidas	87
Capítulo 4:	Análisis del sistema de información	90
4.1.	Definición del sistema	91
4.1.1.	Determinación del Alcance del sistema	91
4.2.	ASI 2: Establecimiento de requisitos	92
4.2.1.	Requisitos funcionales	92



4.2.1.1.	Registro de usuario.....	92
4.2.1.1.	Inicio de sesión	93
4.2.1.2.	Cerrar sesión.....	94
4.2.1.3.	Grabar una actividad	94
4.2.1.4.	Gestionar una actividad realizada	96
4.2.1.5.	Visualización de métricas generales.....	96
4.2.1.6.	Predicciones de tiempos	98
4.2.1.7.	Visualización de un histórico	98
4.2.2.	Requisitos no funcionales	99
4.2.2.1.	Tecnológicos	99
4.2.2.2.	Usabilidad.....	99
4.2.2.3.	Seguridad.....	100
4.2.2.4.	Tiempos de respuesta	100
4.3.	Identificación de Actores del Sistema	100
4.3.1.	Usuario no registrado/ no identificado	100
4.3.2.	Usuario registrado / identificado	100
4.4.	Especificación de Casos de Uso.....	100
4.4.1.	Casos de uso para usuario no registrado	101
4.4.2.	Casos de uso para usuario identificado.....	102
4.4.2.1.	Grabar actividades.....	102
4.4.2.2.	Visualizar los datos diarios	103
4.4.2.3.	Mostrar historial.....	104
4.4.2.4.	Gestionar actividades	105
4.4.2.5.	Consultar estimaciones de tiempo.....	106
4.5.	ASI 3: Identificación de Subsistemas de Análisis.....	106
4.5.1.	Descripción de los subsistemas.....	106
4.5.1.1.	Subsistema de grabación y visión de actividades.....	106
4.5.1.2.	Subsistema de métricas y datos diarios	106
4.5.1.3.	Subsistema de predicciones	107
4.5.1.4.	Subsistema de identificación.....	107
4.5.1.5.	Subsistema de visión del histórico	107
4.5.2.	Descripción de los Interfaces entre Subsistemas.....	107
4.6.	ASI 4: Análisis de los casos de uso.....	108
4.6.1.	Caso de uso 1: Registro de usuario	108
4.6.2.	Caso de uso 2: Inicio de sesión.....	108
4.6.3.	Caso de uso 3: Grabar una actividad usando los sensores del dispositivo.	109



4.6.4.	Caso de uso 4: Grabar una actividad de manera manual.	110
4.6.5.	Caso de uso 5: Realizar un seguimiento de los pasos realizados	110
4.6.6.	Caso de uso 6: Realizar un seguimiento de las calorías diarias.....	111
4.6.7.	Caso de uso 7: Visualizar y modificar los datos personales	111
4.6.8.	Caso de uso 8: Acceder a los detalles de un día pasado	112
4.6.9.	Caso de uso 9: Mostrar los detalles de una actividad grabada manualmente.	112
4.6.10.	Caso de uso 10: Mostrar los detalles de una actividad grabada con los sensores del dispositivo.	113
4.6.11.	Caso de uso 11: Eliminar una actividad.....	113
4.6.12.	Caso de uso 12: Consultar las predicciones de tiempo.....	114
4.7.	ASI 5: Análisis de clases	114
4.7.1.	Descripción de las clases	115
4.7.1.1.	Clases comunes	115
4.7.1.2.	Subsistema de métricas y datos diarios	115
4.7.1.3.	Subsistema de identificación.....	116
4.7.1.4.	Subsistema de predicciones	116
4.7.1.5.	Subsistema de visión del histórico	117
4.7.1.6.	Subsistema de grabación y visión de actividades.....	117
4.8.	ASI 8: Definición de interfaces de usuario	119
4.8.1.	Descripción de la interfaz.....	120
4.8.1.1.	Prototipo de pantalla de registro	120
4.8.1.2.	Prototipo de pantalla de inicio de sesión	120
4.8.1.3.	Prototipo de pantalla principal.....	121
4.8.1.4.	Prototipo de pantalla de grabación automática.....	122
4.8.1.5.	Prototipo de pantalla de grabación manual.....	122
4.8.1.6.	Prototipo de pantalla de visión de actividades automáticas.....	123
4.8.1.7.	Prototipo de pantalla de visión de actividades manuales.....	123
4.8.1.8.	Prototipo de pantalla de visión del histórico.....	124
4.8.1.9.	Prototipo de pantalla de visión de un día histórico.....	125
4.8.1.10.	Prototipo de pantalla de predicciones	125
4.8.2.	Diagrama de navegabilidad	126
4.9.	Especificación del plan de pruebas	126
4.9.1.	Pruebas unitarias.....	127
4.9.2.	Pruebas de integración.....	127
4.9.3.	Pruebas de usabilidad	127
Capítulo 5:	Diseño del Sistema de Información	129



5.1.	DSI 3: Diseño de casos de uso reales	130
5.1.1.	Caso de uso 3: Grabar una actividad usando los sensores del dispositivo	130
5.1.1.1.	Diagrama de Secuencia	130
5.1.2.	Caso de uso 4: Grabar una actividad de manera manual.	131
5.1.2.1.	Diagrama de secuencia.....	131
5.1.3.	Caso de uso 4: Realizar un seguimiento de los pasos realizados.	132
5.1.3.1.	Diagrama de secuencia.....	132
5.1.4.	Caso de uso 9: Mostrar los detalles de una actividad grabada manualmente.	133
5.1.4.1.	Diagrama de secuencia.....	133
5.1.5.	Caso de uso 11: Eliminar una actividad.....	134
5.1.5.1.	Diagrama de secuencia.....	134
5.2.	DSI 4: Diseño de clases	135
5.2.1.	Diagrama de clases.....	135
5.2.2.	Descripción de las clases	140
5.2.2.1.	Utils.....	140
5.2.2.2.	Dto	140
5.2.2.3.	Controller.....	143
5.2.2.3.1.	Fragment	146
5.2.2.3.2.	Dialog	148
5.2.2.4.	Model	150
5.3.	DSI 5: Diseño de la arquitectura de módulos del sistema.....	153
5.3.1.	DSI 5.1 Diseño de Módulos del Sistema	153
5.3.1.1.	Diagrama de paquetes	153
5.3.1.1.1.	Paquete com.example.trainingPartner.controller.dialog.....	155
5.3.1.1.2.	Paquete com.example.trainingPartner.controller.fragment.....	155
5.3.1.1.3.	Paquete com.example.trainingPartner.controller.service	156
5.3.1.1.4.	Paquete com.example.trainingPartner.model.....	156
5.3.1.1.5.	Paquete com.example.trainingPartner.dto	156
5.3.1.1.6.	Paquete com.example.trainingPartner.utils	157
5.3.1.1.7.	Paquete res.drawable.....	157
5.3.1.1.8.	Paquete res.font	157
5.3.1.1.9.	Paquete res.layout.....	158
5.3.1.1.10.	Paquete res.menu	158
5.3.1.1.11.	Paquete res.mipmap	158
5.3.1.1.12.	Paquete res.values	159
5.3.1.1.13.	Paquete res.xml.....	159



5.3.2.	DSI 5.2 Diseño de Comunicaciones entre Módulos	159
5.3.2.1.	Diagrama de componentes	159
5.3.2.2.	Diagrama de despliegue	160
5.3.3.	DSI 5.3 Revisión de la Interfaz de Usuario.....	161
5.3.3.1.	Pantalla de carga.	161
5.3.3.2.	Pantalla registro.....	162
5.3.3.3.	Pantalla de logIn.	163
5.3.3.4.	Pantalla principal	164
5.3.3.5.	Pantalla de grabación de actividades con los sensores del dispositivo.....	167
5.3.3.6.	Pantalla de grabado manual.....	168
5.3.3.7.	Pantalla de visión de una actividad automática.	169
5.3.3.8.	Pantalla de visión de una actividad manual.	171
5.3.3.9.	Pantalla de historial.	172
5.3.3.10.	Pantalla de visualización de un día histórico.	173
5.3.3.11.	Pantalla de predicciones.....	174
5.3.3.12.	Carga de datos.	176
5.4.	DSI 6: Diseño físico de datos	176
5.4.1.	Descripción del SGBD usado.....	177
5.4.2.	Documentos	177
5.5.	DSI 10: Especificación técnica del plan de pruebas.....	178
5.5.1.	Pruebas unitarias.....	179
5.5.1.1.	Registro de usuario.....	179
5.5.1.2.	Inicio de sesión	179
5.5.1.3.	Actividad principal.....	180
5.5.1.4.	Grabación con los sensores del dispositivo.....	180
5.5.1.5.	Grabación manual	180
5.5.1.6.	Fragmento con información del usuario	181
5.5.1.7.	Fragmento del historial	181
5.5.1.8.	Fragmento de predicciones.....	181
5.5.1.9.	Detalles de una actividad automática	182
5.5.1.10.	Detalles de una actividad manual	182
5.5.1.11.	Métodos útiles.....	183
5.5.2.	Pruebas de integración.....	183
5.5.2.1.	Registro de usuario.....	183
5.5.2.2.	Inicio de sesión	184
5.5.2.3.	Actividad principal.....	184



5.5.2.4.	Grabar actividad con los sensores del dispositivo	185
5.5.2.5.	Grabar actividad manualmente.....	185
5.5.2.6.	Fragmento con información del usuario	186
5.5.2.7.	Fragmento del historial	186
5.5.2.8.	Fragmento de un día histórico.....	186
5.5.2.9.	Fragmento de predicciones.....	187
5.5.2.10.	Detalles de una actividad automática	187
5.5.2.11.	Detalles de una actividad manual	187
5.5.2.12.	Pruebas adicionales.....	188
5.5.2.11.1.	Prueba de pasos.....	188
5.5.2.11.2.	Prueba de grabado de la actividad.....	188
5.5.2.11.3.	Prueba de predicciones.....	189
5.5.3.	Pruebas de usabilidad	189
5.5.3.1.	Actividades guiadas.....	189
5.5.3.2.	Cuestionario.	189
5.5.3.2.1.	Preguntas sobre el usuario.....	190
5.5.3.2.2.	Preguntas y anotaciones sobre las actividades guiadas.....	190
5.5.3.2.3.	Cuestionario de usabilidad.....	190
5.5.3.3.	Cuestionario para el responsable de pruebas.....	191
Capítulo 6:	Construcción del sistema de información	193
6.1.	CSI 1: Preparación del entorno de generación y construcción	194
6.1.1.	Estándares y normas seguidos	194
6.1.1.1.	Java Code Conventions.....	194
6.1.1.2.	Arquitectura MVC.....	194
6.1.2.	Lenguajes empleados.....	194
6.1.2.1.	Java	195
6.1.2.2.	XML.....	195
6.1.3.	Librerías y frameworks	195
6.1.3.1.	Material Calendar View	195
6.1.3.2.	Mapbox.....	195
6.1.3.3.	Image-cropper	196
6.1.3.4.	Espresso testing.....	196
6.1.3.5.	Fragment-testing	196
6.1.4.	Herramientas y programas usados para el desarrollo	196
6.1.4.1.	Android Studio.....	196
6.1.4.2.	Cloud Firestore	197



6.1.4.3.	Cloud Storage	197
6.1.4.4.	Free Logo Maker	198
6.1.5.	Herramientas y programas usados para la documentación	198
6.1.5.1.	Word	198
6.1.5.2.	Visio	198
6.1.5.3.	Excel	199
6.1.5.4.	Project	199
6.1.5.5.	Balsamiq	199
6.1.5.6.	Google Forms	200
6.2.	CSI 2: Generación del código de los componentes y procedimientos	200
6.2.1.	Grabado de actividades con los sensores del dispositivo.	201
6.2.2.	Toma de pasos y cálculo de calorías	203
6.3.	CSI 3: Ejecución de las pruebas unitarias	206
6.3.1.	Registro de usuario.....	206
6.3.2.	Inicio de sesión	207
6.3.3.	Actividad principal.....	207
6.3.4.	Grabación con los sensores del dispositivo.....	207
6.3.5.	Grabación manual	208
6.3.6.	Fragmento con información del usuario	208
6.3.7.	Fragmento del historial	208
6.3.8.	Fragmento de predicciones.....	209
6.3.9.	Detalles de una actividad automática	209
6.3.10.	Detalles de una actividad manual	210
6.3.11.	Métodos útiles	210
6.3.12.	Resultado de las ejecuciones	210
6.3.13.	Problemas y errores	212
6.4.	CSI 4: Ejecución de las pruebas de integración	213
6.4.1.	Registro de usuario.....	213
6.4.2.	Inicio de sesión	214
6.4.3.	Actividad principal.....	215
6.4.4.	Grabar actividades con los sensores del dispositivo	215
6.4.5.	Grabar actividad manualmente	216
6.4.6.	Fragmento con información del usuario	216
6.4.7.	Fragmento del historial	217
6.4.8.	Fragmento de un día histórico	217
6.4.9.	Fragmento de predicciones.....	218



6.4.10.	Detalles de una actividad automática	218
6.4.11.	Detalles de una actividad manual	219
6.4.12.	Pruebas adicionales.....	219
6.4.12.1.	Prueba de pasos	219
6.4.12.2.	Prueba de grabado de la actividad.....	221
6.4.12.2.1.	Actividad 1.....	221
6.4.12.2.1.1.	Mapa	221
6.4.12.2.1.2.	Datos generales.....	221
6.4.12.2.1.3.	Parciales	222
6.4.12.2.1.4.	Conclusión.....	222
6.4.12.2.2.	Actividad 2.....	223
6.4.12.2.2.4.	Mapa	223
6.4.12.2.2.5.	Datos generales.....	223
6.4.12.2.2.6.	Parciales	224
6.4.12.2.2.7.	Conclusión.....	224
6.4.12.2.3.	Actividad 3.....	224
6.4.12.2.3.4.	Mapa	224
6.4.12.2.3.5.	Datos generales.....	224
6.4.12.2.3.6.	Parciales	225
6.4.12.2.3.7.	Conclusión.....	225
6.4.12.2.4.	Conclusión de las pruebas de grabado.....	226
6.4.12.3.	Prueba de predicciones	226
6.4.13.	Resultados de las ejecuciones.....	227
6.4.14.	Problemas y errores	229
6.5.	CSI 5: Ejecución de pruebas del sistema	230
6.5.1.	Ejecución de las pruebas de usabilidad.....	230
6.5.1.1.	Pruebas sobre el usuario	230
6.5.1.2.	Actividades guiadas.....	231
6.5.1.3.	Cuestiones de usabilidad.....	235
6.5.1.4.	Correcciones.....	237
6.6.	CSI 6: Elaboración de los manuales de usuario	239
6.6.1.	Manual de instalación	239
6.6.2.	Manual de ejecución	239
6.6.3.	Manual de Usuario	240
6.6.3.1.	Registro e inicio de sesión	240
6.6.3.2.	Cerrar sesión.....	241



6.6.3.3.	Cambiar foto de perfil	242
6.6.3.4.	Cambiar peso y altura.....	243
6.6.3.5.	Grabar una actividad manual	244
6.6.3.6.	Grabar una actividad con los sensores del dispositivo.....	244
6.6.3.7.	Mostrar los detalles de una actividad	246
6.6.3.8.	Visualizar los pasos y calorías.....	248
6.6.3.9.	Visualización de un día histórico	249
6.6.3.10.	Visualización de las predicciones	250
6.6.4.	Manual del programador	251
6.6.4.1.	Android Manifest.....	251
6.6.4.2.	Archivo gradle.....	251
6.6.4.3.	Ejecución de las pruebas	252
6.6.4.4.	Ejecución de la aplicación.....	252
Capítulo 7:	Apéndices.....	254
7.1.	Plan de gestión de riesgos.....	255
7.1.1.	Metodología	255
7.1.2.	Categorías de un riesgo.....	256
7.1.3.	Probabilidad e impacto	256
7.1.4.	Estrategias frente a los riesgos.....	257
7.1.5.	Identificación de los riesgos	257
7.1.6.	Lista de riesgos priorizada	257
7.2.	Conclusiones.....	261
7.2.1.	Conclusiones del proyecto	261
7.2.2.	Conclusiones personales.....	261
7.3.	Ampliaciones	262
7.3.1.	Visión continua de un mapa durante las actividades.....	262
7.3.2.	Posibilidad de compartir las actividades	262
7.3.3.	Visión de actividades en dashboard web.....	263
7.4.	Problemas encontrados	263
7.4.1.	El servicio se paraba después de un tiempo en segundo plano	263
7.4.2.	Manejar SharedPreferences.....	264
7.5.	Referencias bibliográficas	265
7.6.	Contenidos entregados en los anexos	270
7.6.1.	Contenidos	270
7.6.1.1.	Estructura del archivo entregado	270
7.6.1.2.	Estructura del directorio de desarrollo	270



7.6.1.3.	Estructura del directorio de anexos.....	270
7.7.	GNU Free Documentation license.....	271
7.7.0.	PREAMBLE	271
7.7.1.	APPLICABILITY AND DEFINITIONS.....	271
7.7.2.	VERBATIM COPYING.....	273
7.7.3.	COPYING IN QUANTITY.....	273
7.7.4.	MODIFICATIONS.....	273
7.7.5.	COMBINING DOCUMENTS	275
7.7.6.	COLLECTIONS OF DOCUMENTS.....	275
7.7.7.	AGGREGATION WITH INDEPENDENT WORKS.....	276
7.7.8.	TRANSLATION.....	276
7.7.9.	TERMINATION	276
7.7.10.	FUTURE REVISIONS OF THIS LICENSE	277
7.7.11.	RELICENSING	277
7.7.12.	ADDENDUM: How to use this License for your documents.....	278



ÍNDICE DE TABLAS

Tabla 1. Equipos de trabajo y usuarios.....	33
Tabla 2. Comparación con sistemas similares.....	39
Tabla 3. Planificación general.....	60
Tabla 4. Planificación inicial de la fase de planificación.....	60
Tabla 5. Planificación inicial de la fase de análisis.....	61
Tabla 6. Planificación inicial de la fase de diseño.....	62
Tabla 7. Planificación inicial de la fase de implementación.....	63
Tabla 8. Planificación inicial de la fase de documentación.....	63
Tabla 9. Planificación inicial de la fase de cierre del proyecto.....	63
Tabla 10. Riesgos identificados.....	67
Tabla 11. Precio/h del personal.....	68
Tabla 12. Partida de planificación.....	70
Tabla 13. Partida de análisis.....	71
Tabla 14. Partida de diseño.....	72
Tabla 15. Partida de implementación.....	74
Tabla 16. Partida de documentación.....	75
Tabla 17. Partida de cierre del proyecto.....	76
Tabla 18. Partida de otros gastos.....	76
Tabla 19. Resumen del presupuesto de costes.....	77
Tabla 20. Datos para el presupuesto de Cliente.....	78
Tabla 21. Presupuesto de Cliente.....	78
Tabla 22. Bitácora.....	80
Tabla 23. Riesgo: Poca experiencia en el desarrollo de aplicaciones móviles.....	81
Tabla 24. Riesgo: No tener muy clara la funcionalidad del sistema.....	81
Tabla 25. Suspense de alguna asignatura.....	81
Tabla 26. Riesgo: Falta de tiempo por el trabajo.....	81
Tabla 27. Riesgo: Mala climatología.....	81
Tabla 28. Riesgo: Poca experiencia con el desarrollo de pruebas.....	82
Tabla 29. Planificación final general.....	83
Tabla 30. Planificación final de planificación del sistema.....	84
Tabla 31. Planificación final de análisis del sistema.....	84
Tabla 32. Planificación final de diseño del sistema.....	85
Tabla 33. Planificación final de implementación.....	85
Tabla 34. Planificación final de documentación.....	86
Tabla 35. Planificación final de cierre del proyecto.....	86
Tabla 36. Presupuesto final de costes.....	86
Tabla 37. Datos para el cálculo de presupuesto final de Cliente.....	86
Tabla 38. Presupuesto final de Cliente.....	87
Tabla 39. Comparativa de presupuesto final e inicial.....	87
Tabla 40. Caso de uso registro de usuario.....	102
Tabla 41. Caso de uso inicio de sesión.....	102
Tabla 42. Caso de uso grabar actividad con los sensores.....	102
Tabla 43. Caso de uso grabar actividad manual.....	103
Tabla 44. Caso de uso realizar seguimientos pasos.....	103
Tabla 45. Caso de uso realizar seguimiento calorías.....	103



Tabla 46. Caso de uso visualizar y modificar datos personales	103
Tabla 47. Caso de uso visualizar días de uso	104
Tabla 48. Caso de uso acceder a los detalles de un día pasado	104
Tabla 49. Caso de uso mostrar los detalles actividad manual	105
Tabla 50. Caso de uso mostrar los detalles actividad automática	105
Tabla 51. Caso de uso eliminar actividad	105
Tabla 52. Caso de uso consultar predicciones de tiempo	106
Tabla 53. Análisis de caso de uso registro de usuario	108
Tabla 54. Análisis de caso de uso de inicio de sesión	109
Tabla 55. Análisis de caso de uso de grabar actividad con los sensores	109
Tabla 56. Análisis de casos de uso de grabar actividad manual	110
Tabla 57. Análisis de caso de uso de realizar seguimiento pasos	110
Tabla 58. Análisis de caso de uso de realizar seguimiento calorías	111
Tabla 59. Análisis de caso de uso de visualizar los datos personales	111
Tabla 60. Análisis de caso de uso de acceder a los detalles de un día pasado	112
Tabla 61. Análisis de caso de uso de mostrar los detalles de una actividad manual	112
Tabla 62. Análisis de caso de uso de mostrar los detalles de una actividad automática	113
Tabla 63. Análisis de caso de uso de eliminar una actividad	114
Tabla 64. Análisis de caso de uso de consultar las predicciones de tiempo	114
Tabla 65. Descripción DTO	115
Tabla 66. Descripción ModelosBD	115
Tabla 67. Descripción MainWindowActivity	115
Tabla 68. Descripción StepService	116
Tabla 69. Descripción IdentificationActivity	116
Tabla 70. Descripción EstimationActivity	117
Tabla 71. Descripción HistoryActivity	117
Tabla 72. Descripción HistoryInformationActivity	117
Tabla 73. Descripción AutomaticRecorderActivity	118
Tabla 74. Descripción RecorderService	118
Tabla 75. Descripción ManualRecorderActivity	119
Tabla 76. Descripción AutomaticVisorActivity	119
Tabla 77. Descripción ManualVisorActivity	119
Tabla 78. Descripción de la clase Utils.	140
Tabla 79. Descripción de la clase ListAdapter.	140
Tabla 80. Descripción de la clase ActivityDTO	141
Tabla 81. Descripción de la clase LocationActivityDTO	141
Tabla 82. Descripción de la clase ManualActivityDTO.	141
Tabla 83. Descripción de la clase DailyInformationDTO.	141
Tabla 84. Descripción de la clase FastestKilometersDTO	142
Tabla 85. Descripción de la clase PointDTO.	142
Tabla 86. Descripción de la clase UserInformationDTO	143
Tabla 87. Descripción de la clase LoginActivity	143
Tabla 88. Descripción de la clase SignInActivity	144
Tabla 89. Descripción de la clase SplashActivity.	144
Tabla 90. Descripción de la clase ManualRecordActivity	144
Tabla 91. Descripción de la clase REcordActivity.	145
Tabla 92. Descripción de la clase TrainDetailsActivity.	145
Tabla 93. Descripción de la clase TrainDetailsManualActivity	146



Tabla 94. Descripción de la clase MainActivity.	146
Tabla 95. Descripción de la clase DatePickerFragment.....	146
Tabla 96. Descripción de la clase MainWindowFragment.	147
Tabla 97. Descripción de la clase HistoryFragment.	147
Tabla 98. Descripción de la clase HistoricalDayFragment.....	147
Tabla 99. Descripción de la clase PredictionsFragment.....	148
Tabla 100. Descripción de la clase ChangeDataDialog.....	148
Tabla 101. Descripción de la clase ChangeWeightDialog.....	148
Tabla 102. Descripción de la clase ChangeHeightDialog.....	149
Tabla 103. Descripción de la clase DeleteDialog.....	149
Tabla 104. Descripción de la clase SelectModeDialog.....	149
Tabla 105. Descripción de la clase MonthYearDialog.....	149
Tabla 106. Descripción de la clase SignOutDialog.....	150
Tabla 107. Descripción de la clase Model.....	150
Tabla 108. Descripción de la clase DailyInformationModel.....	150
Tabla 109. Descripción de la clase HistoryModel.....	151
Tabla 110. Descripción de la clase LogInSignInModel.....	151
Tabla 111. Descripción de la clase ActivitiesModel.....	151
Tabla 112. Descripción de la clase PredictionsModel.....	152
Tabla 113. Descripción de la clase RecordModel.....	152
Tabla 114. Descripción de la clase TrainDetailsModel.....	153
Tabla 115. Pruebas unitarias registro de usuario.....	179
Tabla 116. Pruebas unitarias inicio de sesión.....	180
Tabla 117. Pruebas unitarias actividad principal.....	180
Tabla 118. Pruebas unitarias grabación con los sensores del dispositivo.....	180
Tabla 119. Pruebas unitarias grabación manual.....	181
Tabla 120. Pruebas unitarias fragmento con información del usuario.....	181
Tabla 121. Pruebas unitarias fragmento del historial.....	181
Tabla 122. Pruebas unitarias fragmento de predicciones.....	182
Tabla 123. Pruebas unitarias detalles de una actividad automática.....	182
Tabla 124. Pruebas unitarias detalles de una actividad manual.....	183
Tabla 125. Pruebas unitarias utils.....	183
Tabla 126. Pruebas de integración registro de usuario.....	184
Tabla 127. Pruebas de integración inicio de sesión.....	184
Tabla 128. Pruebas de integración actividad principal.....	185
Tabla 129. Pruebas de integración grabar actividad con los sensores del dispositivo.....	185
Tabla 130. Pruebas de integración grabar actividad manualmente.....	185
Tabla 131. Pruebas de integración fragmento con información del usuario.....	186
Tabla 132. Pruebas de integración fragmento del historial.....	186
Tabla 133. Pruebas de integración fragmento de un día histórico.....	187
Tabla 134. Pruebas de integración fragmento de predicciones.....	187
Tabla 135. Pruebas de integración detalles de una actividad automática.....	187
Tabla 136. Pruebas de integración detalles de una actividad manual.....	188
Tabla 137. Grado de satisfacción con las predicciones.....	189
Tabla 138. Actividades guiadas.....	189
Tabla 139. Preguntas referentes al usuario.....	190
Tabla 140. Preguntas y anotaciones sobre las actividades guiadas.....	190
Tabla 141. Preguntas de usabilidad.....	191



Tabla 142. Cuestionario para le responsable de pruebas.	191
Tabla 143. Ejecución de las pruebas unitarias: Registro de usuario.	207
Tabla 144. Ejecución de las pruebas unitarias: Inicio de sesión.	207
Tabla 145. Ejecución de las pruebas unitarias: Actividad principal.	207
Tabla 146. Ejecución de las pruebas unitarias: Grabación con los sensores del dispositivo. ..	208
Tabla 147. Ejecución de las pruebas unitarias: Grabación manual.	208
Tabla 148. Ejecución de las pruebas unitarias: Fragmento con información del usuario.	208
Tabla 149. Ejecución de las pruebas unitarias: Fragmento del historial.	209
Tabla 150. Ejecución de las pruebas unitarias: Fragmento de predicciones.	209
Tabla 151. Ejecución de las pruebas unitarias: Fragmento del historial.	209
Tabla 152. Ejecución de las pruebas unitarias: Detalles de una actividad manual.	210
Tabla 153. Ejecución de las pruebas unitarias: : Métodos utils.	210
Tabla 154. Ejecución de las pruebas de integración: Registro de usuario.	214
Tabla 155. Ejecución de las pruebas de integración: Inicio de sesión.	215
Tabla 156. Ejecución de las pruebas de integración: Actividad principal.	215
Tabla 157. Ejecución de las pruebas de integración: Grabar actividades con los sensores del dispositivo.	216
Tabla 158. Ejecución de las pruebas de integración: Grabar actividad manualmente.	216
Tabla 159. Ejecución de las pruebas de integración: Fragmento con información del usuario.	217
Tabla 160. Ejecución de las pruebas de integración: Fragmento del historial.	217
Tabla 161. Ejecución de las pruebas de integración: Fragmento de un día histórico.	218
Tabla 162. Ejecución de las pruebas de integración: Fragmento de predicciones.	218
Tabla 163. Ejecución de las pruebas de integración: Detalles de una actividad automática. .	219
Tabla 164. Ejecución de las pruebas de integración: Detalles de una actividad manual.	219
Tabla 165. Prueba de pasos 1.	220
Tabla 166. Prueba de pasos 2.	220
Tabla 167. Prueba de pasos 3.	221
Tabla 168. Mapa actividad 1.	221
Tabla 169. Datos generales actividad 1.	222
Tabla 170. Parciales actividad 1.	222
Tabla 171. Mapa actividad 2.	223
Tabla 172. Datos generales actividad 2.	223
Tabla 173. Parciales actividad 2.	224
Tabla 174. Mapa actividad 3.	224
Tabla 175. Datos generales actividad 3.	225
Tabla 176. Parciales actividad 3.	225
Tabla 177. Prueba de predicciones usuario 1.	226
Tabla 178. Grado de satisfacción usuario 1.	226
Tabla 179. Prueba de predicciones usuario 2.	226
Tabla 180. Grado de satisfacción usuario 2.	227
Tabla 181. Prueba de predicciones usuario 3.	227
Tabla 182. Grado de satisfacción usuario 3.	227
Tabla 183. Ejecución de las pruebas de usabilidad: Actividades guiadas.	234
Tabla 184. Lista de riesgos priorizada.	260
Tabla 185. Estructura del archivo entregado.	270
Tabla 186. Estructura del directorio de desarrollo.	270
Tabla 187. Estructura del directorio de anexos.	270



ÍNDICE DE ILUSTRACIONES

Ilustración 1. Acelerómetro.....	34
Ilustración 2. Estilo de un mapa.....	36
Ilustración 3. Garmin Connect.....	37
Ilustración 4. Strava.....	37
Ilustración 5. Polar Flow.....	38
Ilustración 6. Java.....	43
Ilustración 7. Android Studio.....	43
Ilustración 8. Swift.....	44
Ilustración 9. Xamarin.....	44
Ilustración 10. Base de datos relacional.....	45
Ilustración 11. MongoDB.....	46
Ilustración 12. Firebase.....	47
Ilustración 13. OBS.....	51
Ilustración 14. PBS Estructura general.....	52
Ilustración 15. PBS Planificación del sistema.....	52
Ilustración 16. PBS Análisis.....	53
Ilustración 17. PBS Diseño.....	53
Ilustración 18. PBS Desarrollo.....	54
Ilustración 19. PBS Documentación.....	54
Ilustración 20. PBS Cierre del proyecto.....	55
Ilustración 21. WBS Estructura general.....	55
Ilustración 22. WBS Planificación del sistema.....	56
Ilustración 23. WBS Análisis.....	56
Ilustración 24. WBS Diseño.....	57
Ilustración 25. WBS Implementación.....	58
Ilustración 26. WBS Documentación.....	59
Ilustración 27. WBS Cierre del proyecto.....	59
Ilustración 28. Diagrama de Gantt General.....	64
Ilustración 29. Diagrama de Gantt planificación.....	64
Ilustración 30. Diagrama de Gantt de análisis.....	64
Ilustración 31. Diagrama de Gantt de diseño.....	65
Ilustración 32. Diagrama de Gantt de desarrollo.....	65
Ilustración 33. Diagrama de Gantt de documentación.....	65
Ilustración 34. Diagrama de Gantt de cierre.....	66
Ilustración 35. Resumen de la funcionalidad del sistema.....	101
Ilustración 36. Casos de uso para usuario no registrado.....	101
Ilustración 37. Casos de uso grabar actividades.....	102
Ilustración 38. Visualizar los datos diarios.....	103
Ilustración 39. Casos de uso mostrar historial.....	104
Ilustración 40. Casos de uso gestionar actividades.....	105
Ilustración 41. Casos de uso consultar estimaciones de tiempo.....	106
Ilustración 42. Comunicación entre subsistemas.....	107
Ilustración 43. Diagrama de clases preliminar.....	114



Ilustración 44. Prototipo pantalla de registro	120
Ilustración 45. Prototipo pantalla de inicio de sesión	121
Ilustración 46. Prototipo pantalla principal.....	121
Ilustración 47. Prototipo pantalla de grabación automática	122
Ilustración 48. Prototipo pantalla de grabación manual.....	123
Ilustración 49. Prototipo pantalla de visión de actividades automáticas	123
Ilustración 50. Prototipo pantalla de visión de actividades manuales.....	124
Ilustración 51. Prototipo de pantalla de visión del histórico	124
Ilustración 52. Prototipo de visión de un día histórico	125
Ilustración 53. Prototipo de pantalla de predicciones	125
Ilustración 54. Diagrama de navegabilidad	126
Ilustración 55. Diagrama de secuencia grabar una actividad con los sensores del dispositivo.131	
Ilustración 56. Diagrama de secuencia grabar una actividad manual.	132
Ilustración 57. Diagrama de secuencia realizar un seguimiento de los pasos realizados.....	133
Ilustración 58. Diagrama de secuencia de mostrar una actividad grabada manualmente.....	134
Ilustración 59. Diagrama de secuencia eliminar una actividad.....	135
Ilustración 60. Diagrama de clases: comunicación entre paquetes.....	136
Ilustración 61. Diagrama de clases: Paquete Utils.	136
Ilustración 62. Diagrama de clases: Paquete dto.	137
Ilustración 63. Diagrama de clases: Paquete controller.....	138
Ilustración 64. Diagrama de clases: Paquete model	139
Ilustración 65. Diagrama de paquetes.	154
Ilustración 66. Contenido trainingPartner.controller.....	155
Ilustración 67. Contenido trainingPartner.controller.dialog.....	155
Ilustración 68. Contenido trainingPartner.controller.fragment.....	155
Ilustración 69. Contenido trainingPartner.controller.service.	156
Ilustración 70. Contenido trainingPartner.model.	156
Ilustración 71. Contenido trainingPartner.dto.....	156
Ilustración 72. Contenido trainingPartner.utils.....	157
Ilustración 73. Contenido res.drawable.	157
Ilustración 74. Contenido res.font.	158
Ilustración 75. Contenido res.layout.	158
Ilustración 76. Contenido res.menu.....	158
Ilustración 77. Contenido res.mipmap.....	159
Ilustración 78. Contenido res.values.....	159
Ilustración 79. Contenido res.xml.	159
Ilustración 80. Diagrama de componentes.	160
Ilustración 81. Diagrama de despliegue.....	161
Ilustración 82. Pantalla de carga.	162
Ilustración 83. Pantalla de registro.	163
Ilustración 84. Muestra de error.....	163
Ilustración 85. Pantalla de Login.	164
Ilustración 86. Pantalla principal.....	165
Ilustración 87. Imagen usuario sin foto.....	165
Ilustración 88. Interfaz cambio de altura	166
Ilustración 89. Interfaz cambio de peso.....	166
Ilustración 90. Notificación pasos.	166
Ilustración 91. Elemento automático.....	166



Ilustración 92. Elemento manual.	166
Ilustración 93. Interfaz cerrar sesión.....	167
Ilustración 94. Pantalla de grabación con los sensores.....	167
Ilustración 95. Interfaz elección de modo.....	168
Ilustración 96. Notificación ubicación.	168
Ilustración 97. Pantalla de grabación manual.....	169
Ilustración 98. Pantalla de visión de actividad automática.....	170
Ilustración 99. Mapa con paradas.....	170
Ilustración 100. Mapa sin puntos.....	171
Ilustración 101. Interfaz borrar actividad.....	171
Ilustración 102. Pantalla de visión de actividad manual.....	172
Ilustración 103. Pantalla de visión del historial.....	173
Ilustración 104. Dialog selector de fecha.....	173
Ilustración 105. Pantalla de visualización de un día histórico.....	174
Ilustración 106. Pantalla de predicciones.....	175
Ilustración 107. Pantalla de predicciones con datos.....	176
Ilustración 108. Carga de datos.....	176
Ilustración 109. Documentos base de datos.....	177
Ilustración 110. Diagrama base de datos.....	178
Ilustración 111. Arquitectura MVC.....	194
Ilustración 112. Java.....	195
Ilustración 113. XML.....	195
Ilustración 114. Mapbox.....	196
Ilustración 115. Espresso.....	196
Ilustración 116. Android studio.....	197
Ilustración 117. Cloud Firestore.....	197
Ilustración 118. Google Cloud Storage.....	197
Ilustración 119. Free LogoMaker.....	198
Ilustración 120. Word.....	198
Ilustración 121. Visio.....	198
Ilustración 122. Excel.....	199
Ilustración 123. Project.....	199
Ilustración 124. Balsamiq.....	200
Ilustración 125. Google Forms.....	200
Ilustración 126. Grabado de actividades con los sensores del dispositivo: Permisos.....	201
Ilustración 127. Grabado de actividades con los sensores del dispositivo: Callback.....	201
Ilustración 128. Grabado de actividades con los sensores del dispositivo: onNewLocation....	202
Ilustración 129. Grabado de las actividades con los sensores del dispositivo: onNewLocation 2.	202
Ilustración 130. Grabado de las actividades con los sensores del dispositivo: saveRoute.....	203
Ilustración 131. Toma de pasos y calorías: onSensorChanged.....	204
Ilustración 132. Toma de pasos y calorías: Calcular longitud de paso.....	205
Ilustración 133. Toma de pasos y calorías: Constantes para pasos.....	205
Ilustración 134. Toma de pasos y calorías: updateSteps.....	206
Ilustración 135. Ejecución de las pruebas unitarias: Resultados de las ejecuciones de las pruebas.....	210
Ilustración 136. Ejecución de las pruebas unitarias: Resultado de las ejecuciones de las pruebas 1.....	211



Ilustración 137. Ejecución de las pruebas unitarias: Resultado de las ejecuciones de las pruebas 2.....	211
Ilustración 138. Problema 1.	212
Ilustración 139. Solución al problema 1.	213
Ilustración 140. Pruebas de integración: Resultado de las ejecuciones.	227
Ilustración 141. Pruebas de integración: Resultado de las pruebas 1.	228
Ilustración 142. Pruebas de integración: Resultado de las pruebas 2.	228
Ilustración 143. Pruebas de integración: Resultado de las pruebas 3.	229
Ilustración 144. Problemas pruebas de integración.	229
Ilustración 145. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 1.	230
Ilustración 146. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 2.	230
Ilustración 147. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 3.	231
Ilustración 148. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 4.	231
Ilustración 149. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 5.	231
Ilustración 150. Ejecución de las pruebas de usabilidad: Actividades guiadas 1.	232
Ilustración 151. Ejecución de las pruebas de usabilidad: Actividades guiadas 2.	232
Ilustración 152. Ejecución de las pruebas de usabilidad: Actividades guiadas 4.	235
Ilustración 153. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 1.	235
Ilustración 154. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 2.	235
Ilustración 155. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 3.	236
Ilustración 156. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 4.	236
Ilustración 157. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 5.	236
Ilustración 158. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 6.	237
Ilustración 159. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 7.	237
Ilustración 160. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 8.	237
Ilustración 161. Corrección notificación.	238
Ilustración 162. Corrección colores.	238
Ilustración 163. Corrección barra 1.	238
Ilustración 164. Corrección barra 2.	238
Ilustración 165. Icono de la aplicación.	239
Ilustración 166. Manual de usuario: Registro.	240
Ilustración 167. Manual de usuario: Botón iniciar sesión.	241
Ilustración 168. Manual de usuario: Inicio de sesión.	241
Ilustración 169. Manual de usuario: Botón de registro.	241
Ilustración 170. Manual de usuario: Cerrar sesión.	242
Ilustración 171. Manual de usuario: Cambiar foto de perfil 1.	242
Ilustración 172. Manual de usuario: Cambiar foto de perfil 2.	243
Ilustración 173. Manual de usuario: Cambiar foto de perfil 3.	243
Ilustración 174. Manual de usuario: Cambiar peso y altura.	243
Ilustración 175. Manual de usuario: Grabar actividad manual.	244
Ilustración 176. Manual de usuario: Permisos.	245
Ilustración 177. Manual de usuario: Determinando ubicación.	245
Ilustración 178. Manual de usuario: Grabar actividades con los sensores del dispositivo.	246
Ilustración 179. Manual de usuario: Elementos actividad.	246
Ilustración 180. Manual de usuario: Detalles de una actividad manual.	247
Ilustración 181. Manual de usuario: Detalles de una actividad automática.	248
Ilustración 182. Manual de usuario: Visualización de pasos y calorías.	249
Ilustración 183. Manual de usuario: Visualización de un día histórico.	250



Ilustración 184. Manual de usuario: Seleccionar fecha.	250
Ilustración 185. Manual de usuario: Visualización de las predicciones.	251
Ilustración 186. Archivo gradle.	252
Ilustración 187. Ejecución de las pruebas.	252
Ilustración 188. Ejecución de la aplicación.....	252
Ilustración 189. Metodología 1.	255
Ilustración 190. Metodología 2.	255
Ilustración 191. Categorías de un riesgo.....	256
Ilustración 192. Matriz de probabilidad e impacto.....	257
Ilustración 193. Posibilidad de compartir actividades.	262
Ilustración 194. Dashboard web.	263
Ilustración 195. Sin restricciones de batería.	264



¿QUÉ ES ESTE TRABAJO?

RESUMEN

En este trabajo se desarrolla una aplicación móvil para el seguimiento de métricas de salud de los usuarios y la grabación de sus entrenamientos de caminata o running.

La aplicación muestra datos del usuario (pasos, calorías, peso, altura, imagen), algunos de estos podrán ser cambiados y otros se actualizarán de manera automática a través de los sensores del dispositivo. La aplicación también permite guardar actividades al aire libre, ya sea a través del sensor GPS o de manera manual introduciendo la distancia y el tiempo. La aplicación también calcula predicciones de tiempos en ciertas distancias a partir de datos de los entrenamientos realizados y da la posibilidad de visualizar las métricas tomadas en días pasados.

La tecnología utilizada para realizar todo el proyecto ha sido Java con Android Studio. Todo el guardado de datos como de imágenes se ha hecho a través de los servicios de Google, más concretamente Firebase Firestore.

El objetivo principal del trabajo es crear una aplicación que permita a cualquier persona realizar un seguimiento de su salud de manera sencilla, sin importar el manejo que tenga con dispositivos móviles ni su destreza deportiva.

PALABRAS CLAVE

Palabras clave: Sensores, entrenamientos, segundo plano, métricas.

ABSTRACT

In this project a mobile app is developed to keep up with health metrics of users and record their walking and running trainings.

The app shows users data (steps, calories, weight, height, profile image). Some of these can be changed and others will be automatically updated through the sensors of the device. The app also allows to save outdoor activities, both through GPS sensor or manually, introducing distance and time. The app also calculates predictions of times for certain distances thanks to the trainings carried out and allows for the possibility of seeing the metrics taken in previous days.



The technology used for the development of this project was Java with Android Studio. All the data and images savings were made through Google services, more specifically those of Firebase Firestore.

The main objective in this project is to create an app that enables anyone to execute a follow up on their health in an easy manner, not being relevant their level of knowledge about mobile devices or their physical abilities.

KEYWORDS

Keywords: Sensors, workouts, background, metrics.



Escuela de
Ingeniería Informática
Universidad de Oviedo





CAPÍTULO 1: PLANIFICACIÓN DEL SISTEMA DE INFORMACIÓN

PLANIFICACIÓN

FASE DE

PSI



1.1. PSI 1: INICIO DEL PLAN DE SISTEMAS DE INFORMACIÓN

1.1.1. PSI 1.1: Análisis de la Necesidad del PSI

Nos encontramos en la era de la inmediatez, disponemos de todas las comodidades al alcance de nuestra mano. Podemos alimentarnos, entretenernos, relacionarnos e incluso culturizarnos sin salir de la comodidad de nuestro hogar. Sin embargo, esa cómoda inmediatez esconde un grave problema que se hace más latente cuanto más nos acostumbramos a tenerlo todo con un solo clic.

El cuerpo humano necesita actividad. Así, cada vez más personas padecen las consecuencias del sedentarismo, el sobrepeso y la hipertensión, entre otras. Según la Organización Mundial de la Salud, la actividad física regular reduce el riesgo de muerte prematura y disminuye en un 50% las probabilidades de padecer enfermedades cardiovasculares o fracturas de cadera [1]. Los problemas que acarrea el sedentarismo no sólo son físicos: una vida activa contribuye al bienestar psicológico, reduce el estrés, la ansiedad, la depresión y los sentimientos de soledad.

Cada año mueren 17,1 millones de personas en el mundo a causa de enfermedades cardiovasculares [2]. Según la OMS, el 80% de estas muertes prematuras podrían haberse evitado siguiendo un estilo de vida saludable [2]. Es preocupante saber que los datos abalados por la Encuesta Nacional de Salud de 2017 muestran que un 40,3% de las mujeres se consideran sedentarias, frente a un 32,3% de los hombres, y que durante el tiempo de ocio, solo un 8,4% de los hombres y un 7,0% de mujeres realizan ejercicio diariamente [3]. Estas estadísticas guardan relación con las de La Encuesta Europea de Salud en España del año 2020, que aseguran que, en los grupos de edad de 35 a 74 años, un 44,9% de los hombres y un 30,6% de las mujeres padecen sobrepeso, enfermedad que causa en torno a 131.000 muertes al año en nuestro país [4].

Debido a todos los datos presentados con anterioridad, surge la necesidad de desarrollar una herramienta que, de una forma cómoda y sencilla, permita a sus usuarios a llevar una vida más activa, practicar ejercicio físico a diario y, en definitiva, a moverse. Existen múltiples aplicaciones deportivas como Garmin o Polar, que solo cobran sentido si es un deportista asiduo y se dispone de un reloj inteligente que tome las métricas. El objetivo de este proyecto es crear una aplicación, que permita usando únicamente nuestro móvil realizar algunas de las acciones más básicas para cuidar nuestra salud, todo esto sin confundir al usuario con datos innecesarios. De esta manera se acerca el deporte y la vida sana al público general ya que, según el Instituto Nacional de Estadística, el 99,5% de la población de 16 a 74 años dispone de un dispositivo móvil.

1.1.2. PSI 1.2: Identificación del Alcance del PSI

El presente proyecto persigue desarrollar una aplicación móvil con las siguientes funcionalidades:



- El usuario deberá tener una cuenta única e intransferible a la que acceder con un usuario y contraseña. La aplicación permitirá mantener un histórico de los datos y métricas.
- La aplicación gestionará los datos de seguimiento del usuario. Las métricas que se van a registrar son:
 - Pasos realizados el día actual.
 - Calorías quemadas
 - Entrenamientos realizados en ese día
- Se podrán grabar los entrenamientos. De manera sencilla, se podrá guardar la práctica deportiva realizada. Esto implica que, al acabar la actividad, se podrá visualizar un mapa con: el recorrido realizado, la velocidad media, la distancia en kilómetros que se ha desplazado y el número de calorías quemadas. Además, para cada entrenamiento existirá la opción de adjuntarle imágenes.
- La aplicación hará una estimación (basada en los entrenamientos realizados) de los tiempos que podría realizar el usuario en distancias de: 1km, 10km, media maratón (21km) y maratón (42km)
- La aplicación debe de permitir introducir entrenamientos manualmente, es decir sin necesidad de grabarlos con la aplicación. Debe de ser posible adjuntar el tipo de actividad (caminata o running), la distancia y el tiempo que tardó. Estas actividades, al igual que las otras, pueden ser revisadas en el futuro.

Por tanto, los objetivos estratégicos, para conseguir que la aplicación sea un éxito son:

- Creación de un sistema de inicio de sesión.
- Automatización de la toma de pasos.
- Automatización del cálculo de calorías.
- Visualización de los datos realizados en el día actual y en los pasados.
- Guardar toda la información recabada durante una actividad.
- Proyección mediante los datos históricos, del tiempo que se tardaría en realizar diferentes distancias.
- Introducción manual de la actividad.

1.2. PSI 2: DEFINICIÓN Y ORGANIZACIÓN DEL PSI

1.2.1. PSI 2.1: Especificación del Ámbito y Alcance

En función de los objetivos estratégicos vistos, el proyecto se divide en las siguientes fases/objetivos generales:

Fase 1: Fijación teórica de los objetivos y requisitos de la aplicación.

La documentación irá realizándose y evolucionando a lo largo de toda la aplicación. Se dedicará el inicio de este proyecto para realizar la primera parte de la documentación. Fijando claramente los objetivos y requisitos de funcionalidad, así como documentar las tecnologías y herramientas a utilizar.

Adicionalmente, se empleará esta fase para profundizar en los conceptos teóricos que se necesitan conocer para entender este proyecto.



Objetivos de esta fase:

- Presentación teórica de las necesidades que se pretenden atender con nuestra aplicación.
- Estudio de aplicaciones similares a la que se pretende desarrollar para cubrir las necesidades observadas.
- Crear una base sólida de requisitos y funcionalidades, para a partir de ella crear la aplicación.
- Análisis de las diferentes herramientas de programación a utilizar.
- Comenzar la documentación.

Fase 2: Guardado de una ruta mediante el GPS.

Esta segunda fase se utilizará para crear la funcionalidad de guardar las rutas a través del GPS del móvil. El usuario al desplazarse irá viendo la ruta que ha realizado y su posición en el mapa, además de su velocidad actual, el tiempo que lleva realizando la actividad y la distancia recorrida.

La ruta realizada se marcará con una línea. Si se detiene la actividad, la línea que sigue la ubicación del usuario dejará de dibujarse.

Inicialmente, la grabación solo se llevará a cabo con la pantalla encendida, permitiendo al cliente ver sus datos. Se investigará la opción de ejecutarla con el dispositivo bloqueado.

Objetivos de la fase:

- Mostrar la ubicación actual.
- Guardar el recorrido realizado.
- Presentar las métricas de velocidad, tiempo de actividad y distancia.
- Al finalizar la actividad y guardarla, presentar un resumen de todos los datos.
- Investigar en la opción de guardar la ruta con el dispositivo bloqueado.
- Completar los datos de la actividad con fotos y anotaciones a los entrenamientos.

Fase 3: Toma de métricas diarias y muestreo de datos.

Esta fase consiste en diseñar una pantalla principal que muestre un resumen del día del usuario: Las actividades realizadas si las hay, los pasos realizados, las calorías quemadas, una foto de perfil, el peso, altura.

En esta fase, también se desarrollará un calendario desde el que se podrán acceder a los datos históricos.

Objetivos de la fase:

- Contabilizar los pasos realizados por el usuario.
- Realizar una estimación de las calorías quemadas.
- Creación de un calendario para ver un historial.
- Ficha de la condición física.

Fase 4: Creación de una manera de iniciar sesión.



Cada usuario, solo podrá acceder a sus datos. Se creará un inicio de sesión que, independientemente del dispositivo en el que te conectes, le mostrará tus datos guardados.

Objetivos de la fase:

- Crear e integrar un sistema de autenticación en la aplicación.
- Cargar los entrenamientos y datos guardados del usuario cuando inicia sesión.
- Permitir cerrar sesión al usuario.

Fase 5: Estimación de los tiempos que va a realizar un usuario en diferentes distancias.

A partir de diversos algoritmos con los datos históricos de las actividades realizadas y sus características físicas, se hará una estimación de los tiempos que el usuario podría realizar en varias distancias predefinidas (1 kilómetro, 10 kilómetros, 21kilómetros y maratón).

Si no se tienen datos suficientes para el cálculo de estas estimaciones (debido a la falta de entrenamientos registrados), se debe de notificar al usuario que aún no es posible acceder a esta funcionalidad.

Objetivos de esta fase:

- Seleccionar el mejor algoritmo para el cálculo de las aproximaciones.
- Determinar los datos que son necesarios para crear la funcionalidad.
- Desarrollar el algoritmo.
- Mostrar al usuario los requisitos para acceder a esta funcionalidad (qué entrenamientos debe de realizar).

Fase 6: Guardado de los entrenamientos de manera manual.

Permitir introducir los datos de un entrenamiento de manera manual. Sin haber grabado el entrenamiento con la aplicación, el usuario ha de ser capaz de introducir el tipo de deporte (caminata o running), la distancia recorrida y el tiempo que se ha tardado.

Objetivos de la fase:

- Tener control de que los datos son introducidos correctamente.
- Permitir ver estas actividades, junto a las otras en el histórico.
- Usar estos entrenamientos, al igual que los grabados con la aplicación, para el cálculo de otras distancias.
- Permitir introducir fotos a los entrenamientos.

Fase 7: Cierre de la documentación.

Para finalizar y cerrar el proyecto, se realizará una última fase en la que se revisará y se unificará toda la documentación.

Objetivos de la fase:

- Finalización de la documentación.
- Unión de todas las partes.
- Uniformar el estilo visual de la documentación.
- Revisión de la documentación.

1.2.2. PSI 2.2: Organización del PSI

A partir de las diferentes fases del desarrollo del proyecto, se establece la siguiente organización:

USUARIO	PERFIL	FUNCIÓN
EQUIPO DE SUPERVISIÓN		
TUTOR DEL TFG	Jefe de proyecto	Supervisión de que se han llevado a cabo de manera satisfactoria todas las fases.
EQUIPO DE DESARROLLO		
ALUMNO	Investigador	Búsqueda de las diferentes tecnologías disponibles para llevar a cabo la funcionalidad.
ALUMNO	Analista	Análisis de las diferentes tecnologías disponibles.
ALUMNO	Arquitecto del software	Diseño de la arquitectura del sistema.
ALUMNO	Diseñador del software	Diseñar el software de la aplicación.
ALUMNO	Programador	Encargado del implementar la aplicación.
ALUMNO	Administrador de bases de datos	Creador y gestor de las bases de datos de la aplicación.
EQUIPO DE PRUEBAS Y CARGA DE DATOS		
ALUMNO	Tester	Encargado de desarrollar las pruebas de funcionalidad, las de integración y las de usabilidad. Ha de asegurarse de que la funcionalidad es correcta.
EQUIPO DE DISEÑO		
ALUMNO	Diseñador front-end	Encargado del diseño de la interfaz de usuario.

Tabla 1. Equipos de trabajo y usuarios

1.3. PSI 3: ESTUDIO DE LA INFORMACIÓN RELEVANTE

1.3.1. PSI 3.1: Selección y Análisis de Antecedentes

Antes de realizar cualquier tipo de proyecto, es de vital importancia tener claros los conceptos teóricos del dominio del problema. Se usará esta sección para clarificar y explicar ciertos términos importantes para la comprensión de este trabajo. Posteriormente, se estudiarán algunos sistemas similares al que se pretende crear en este proyecto.

1.3.1.1. Conceptos teóricos

1.3.1.1.1 Caloría

Se puede definir caloría [5] como la cantidad de calor necesaria para elevar un grado la temperatura de un litro de agua a la presión atmosférica normal. Una caloría (cal) equivale exactamente a 4,1868 julios (J), mientras que una kilocaloría (kcal) es exactamente 4.186,8 julios (J). Como nuestro organismo maneja cantidades de energía muy altas, se utilizará la kilocaloría (kcal) que equivale a 1000 calorías.

En el dominio de esta aplicación, interesa conocer las calorías quemadas al practicar un deporte. Evidentemente, no todas las personas pierden la misma cantidad de calorías haciendo ejercicio. Una de las variables a tener en cuenta es su peso: cuantos más kilos tenga, más calorías perderá. Lo mismo sucede con su estado físico, ya que aquellas personas que comienzan a ejercitarse quemarán más calorías porque necesitan usar una mayor carga energética para rendir. La velocidad también influye en el recuento de calorías quemadas según el peso. De promedio, quemará, 0,73 calorías a una velocidad media de 4-6 km/h, es decir, caminando. Si aumenta la velocidad y corre, las cifras aumentan hasta los 1,03 kcals por kilo de peso.

Por lo tanto, para determinar la cantidad exacta de calorías quemadas, tendrás que multiplicar tu peso por 1,03 o 0,73 y la distancia recorrida durante el tiempo que ha corrido o caminado.

1.3.1.1.2 Acelerómetro

Un acelerómetro [6] es un dispositivo que permite realizar la medición de la aceleración. Se llama aceleración, en tanto, al acto y el resultado de acelerar y a la magnitud que hace referencia al cambio de la velocidad en una unidad de tiempo.

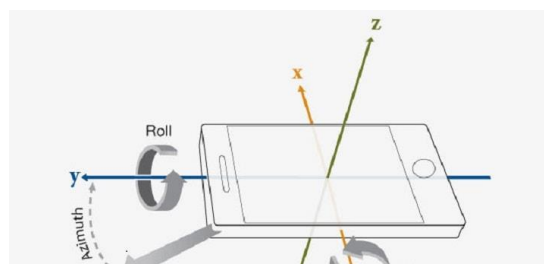


Ilustración 1. Acelerómetro



Un acelerómetro, también puede denominarse sensor de aceleración. Hoy en día prácticamente todos los smartphones disponen de uno, y es el sensor que se va a utilizar en este proyecto para contabilizar los cálculos que hace diariamente el usuario.

1.3.1.1.3 API

El término API [7] es una abreviatura de Application Programming Interfaces. Se trata de un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones, permitiendo la comunicación entre dos aplicaciones de software a través de un conjunto de reglas.

Las API permiten que sus productos y servicios se comuniquen con otros, sin necesidad de saber cómo están implementados. Esto simplifica el desarrollo de las aplicaciones y permite ahorrar tiempo y dinero. Las API le otorgan flexibilidad; simplifican el diseño, la administración y el uso de las aplicaciones.

Las API pueden tener tanto una como varias funciones, de cara a un usuario normal, lo único que vas a ver de una API son los resultados. No son la parte visible, sino los circuitos internos que sólo los desarrolladores ven y conectan para hacer funcionar una herramienta.

1.3.1.1.4 Localización y coordenadas

La localización [8] es la determinación del lugar en el cual se halla una persona, cosa o característica geográfica. Las personas y los objetos están ubicados en un determinado espacio, a esa ubicación se la denomina localización. Para conocer una localización, es necesario una serie de coordenadas. Esta serie de coordenadas permiten establecer: puntos, líneas, polígonos o conjuntos de los anteriores.

Se conoce como sistema de coordenadas al conjunto de los valores que permiten identificar de manera inequívoca la posición de un punto en un espacio. Las coordenadas geográficas, constituyen un subtipo de las denominadas coordenadas esféricas ya que permiten definir puntos sobre la Tierra. El sistema más frecuente es aquel que emplea la latitud y la longitud.

La latitud (norte o sur) y la longitud (este u oeste) permiten conocer los ángulos laterales de la superficie de la Tierra. Ambas coordenadas angulares, se miden desde el centro del planeta y se hallan alineadas con el eje de rotación de la Tierra.

1.3.1.1.5 Estilo o capa de un mapa

Un estilo de un mapa interactivo como pueden ser los de MapBox [9] o GoogleMaps [10], es un documento que define la apariencia visual de un mapa:

- Qué datos representar.
- Cómo se representan estilísticamente estos datos.

Usando estas capas, no solo podemos cambiar la apariencia del mapa (mostrar calles, o imágenes de satélites). A través de ellos podremos dibujar la línea que muestra el recorrido del usuario en el mapa, cambiando también su grosor, color y apariencia.

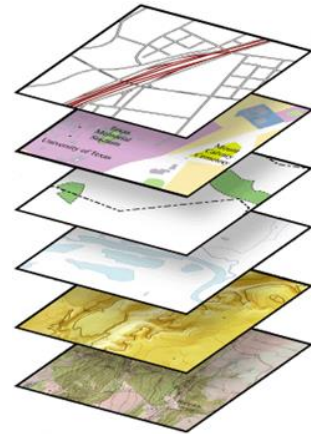


Ilustración 2. Estilo de un mapa

1.3.1.2. Sistemas similares

En este apartado, se van a detallar algunas aplicaciones similares a la propuesta en este trabajo. Posteriormente, se compararán con el proyecto que se va a desarrollar, mostrando los puntos comunes y diferencias

1.3.1.2.1 Garmin connect

DESCRIPCIÓN:

Aplicación de seguimiento del deporte que teniendo acceso a un dispositivo (reloj o pulsera deportiva), da datos de rendimiento. Muestra todos estos datos en su ventana principal, de forma diaria y por medio de tarjetas. Entre los datos que muestra están:

- Frecuencia cardiaca.
- Pasos.
- Calorías.
- Datos del sueño.

Además, permite compartir los entrenamientos con otros usuarios de Garmin.

También pone a disposición del usuario un calendario en el que se puede ver los datos históricos.

PROS:

- Proporciona muchas métricas del usuario debido a el uso de los dispositivos bluetooth.
- El calendario con datos de otros días es muy útil.
- El muestreo de los datos con tarjetas y gráficos es muy visual e intuitivo.
- Permite programar entrenamientos para luego realizarlos en el reloj.

CONTRAS:

- Para usarla, necesitas un dispositivo Garmin que envíe los datos.
- No se pueden introducir entrenamientos usando directamente el móvil.

- Se pueden programar entrenamientos para realizarlos luego con el reloj, pero no se pueden introducir los datos de un entrenamiento que no has grabado.



Ilustración 3. Garmin Connect

1.3.1.2.2 Strava

DESCRIPCIÓN:

Strava es una red social para corredores, permite grabar entrenamientos directamente desde la aplicación y, si has hecho una actividad a través de otra plataforma o dispositivo, como puede ser Garmin o Polar, te da la opción de importarlos. La función principal de Strava es mostrar los datos de los entrenamientos a otros usuarios, que pueden dar “me gusta” y escribir comentarios.

PROS:

- Permite grabar entrenamientos de manera nativa con la aplicación.
- La interacción con otros usuarios es mucho mejor que otras aplicaciones.
- Se pueden guardar fotos junto a los entrenamientos.

CONTRAS:

- Los datos que muestra de los entrenamientos no son muy completos, no entran mucho en detalle
- No muestra métricas del día a día, como pueden ser los pasos o la frecuencia cardiaca fuera del entrenamiento.

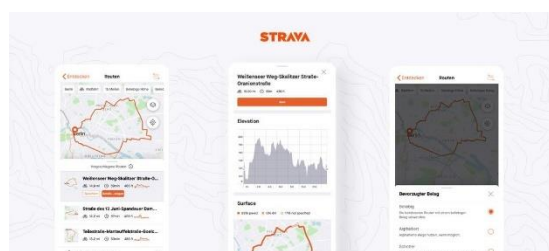


Ilustración 4. Strava

1.3.1.2.3 Polar flow

DESCRIPCIÓN:

Aplicación deportiva de Polar, se centra en la toma de datos por medio de un dispositivo Polar. En este caso también da la opción de grabar las actividades usando el móvil. Da acceso a un calendario con los datos de cada día. También pone a disposición del usuario una serie de blogs sobre deporte, que se pueden consultar en un apartado de la aplicación.

PROS:

- Permite además de grabar entrenamientos con un reloj, hacerlo directamente desde la aplicación
- La introducción de los blogs deportivos
- El acceso a los datos en un calendario

CONTRAS:

- La interfaz es muy poco visual
- El muestreo de los datos no es muy intuitivo
- No se pueden compartir los entrenamientos

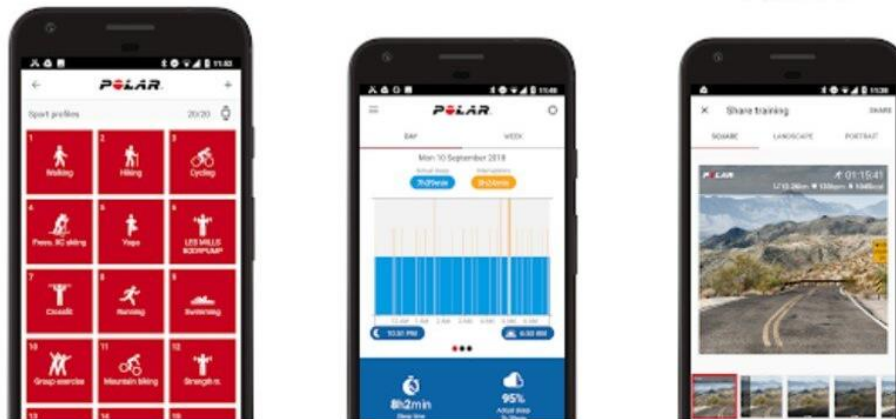


Ilustración 5. Polar Flow

1.3.1.3. Comparación entre sistemas similares

A continuación, a partir de la información detallada en el apartado anterior y en [1.1.2. PSI 1.2: Identificación del Alcance del PSI](#), se hará una comparación de los sistemas similares y de la propuesta. Resaltando las funcionalidades que proporcionan y las que no.

Se marcará con una ✖ los servicios que no se proporcionen y con un ✓ los que, si se satisfacen, y un • si lo hacen en mayor o menor medida.

FUNCIONALIDAD	GARMIN CONNECT	STRAVA	POLAR FLOW	TRAINING PARTNER
Grabar entrenamientos únicamente con el móvil.	✗	✓	✗	✓
Añadir entrenamientos de manera manual.	✗	✓	✗	✓
Mostrar un calendario con las actividades de otros días.	✓	✗	•	✓
Guardar datos fuera de entrenamientos (pasos, calorías, etc.)	✓	✗	•	✓
Compatibilidad con un reloj inteligente para tomar las métricas.	✓	✓	✓	✗
Estimaciones de tiempos en otras distancias.	•	✗	✗	✓

Tabla 2. Comparación con sistemas similares

1.3.1.4. Estudio de APIs relevantes

En este apartado, se describen dos APIs que permiten el muestreo y la interacción con un mapa, para poder rastrear, mostrar y guardar la ubicación del cliente. A continuación, se exponen las características principales de cada una de ellas.

1.3.1.4.1 Google Maps

Google Maps [10] emplea un SDK de uso común. Debido a la gran popularidad de la aplicación de mapas desarrollada por el mismo Google, es una de las APIs de geolocalización más usadas. Cuenta con una amplia base de datos de ubicaciones y rutas. También incluye otras funciones: la geocodificación, la información de localización, las imágenes y el enrutamiento. Entre las ventajas de Google Maps se ha de mencionar que tiene una gran comunidad de desarrolladores, una amplia su popularidad y extenso soporte de idiomas.

Sin embargo, su mayor problema [11] reside en su complejo plan de pagos. La plataforma ofrece una línea de crédito de 200\$ mensuales de los cuales el usuario puede disponer libremente. En el momento en que se termine, se cobrará un importe adicional por cada uso. Adicionalmente, no presenta código abierto y los mapas requieren de conexión para cargarse.

1.3.1.4.2 MapBox

Mapbox [9] cuenta con el apoyo de una comunidad internacional de cartógrafos voluntarios. Se encargan de que los datos sean abiertos y de libre acceso.

Mapbox [11] utiliza diferentes componentes, véase:

- GDAL: Librería de código abierto para trabajar con datos geoespaciales.
- Leaflet: Biblioteca en JavaScript para crear mapas interactivos.
- Mapnik: es una herramienta de código abierto que permite renderizar mapas



Esta API tiene otras ventajas [11], como la posibilidad de usarse sin conexión. Además, gracias a Vector Maps, existe compatibilidad entre navegadores web. El SDK es de código abierto y tiene un sistema de pagos que permite 50.000 cargas web y puede tolerar hasta un total de 25.000 usuarios sin coste alguno.





CAPÍTULO 2: DEFINICIÓN DE LA ARQUITECTURA TECNOLÓGICA

PLANIFICACIÓN

FASE DE

PSI

En esta sección, se identifican las posibles tecnologías a emplear. Al ser esta una aplicación que debe de obtener la localización del usuario mientras se mueve, es claro que debe de ser una aplicación móvil. A continuación, se plantean tres opciones para desarrollarla. Seguidamente, se hará un estudio del tipo de base de datos que emplear para el proyecto.

2.1. PSI 7.1: IDENTIFICACIÓN DE LAS NECESIDADES DE INFRAESTRUCTURA TECNOLÓGICA

2.1.1. Aplicación móvil

2.1.1.1. *Java con Android Studio*

La primera opción es usar Android Studio [12] junto con el lenguaje de programación Java [13].

Java [13] es un lenguaje de programación orientado a objetos que se incorporó al ámbito de la informática en los años noventa. La idea de Java [13] es que pueden implementarse programas con la posibilidad de ejecutarse en cualquier contexto, en cualquier ambiente, siendo así su portabilidad uno de sus principales logros.

Las principales características de esta opción son:

- Amplio conocimiento del lenguaje de programación (ya que se ha visto a lo largo de toda la carrera).
- Cierta familiaridad con el entorno Android Studio.
- Mucho código abierto y disponible debido a su popularidad.
- Lenguaje orientado a objetos.



Ilustración 6. Java



Ilustración 7. Android Studio

2.1.1.2. *IOS Nativo con Swift*

El lenguaje Swift [14] está pensado para desarrollo aplicaciones iOS, y como tal, está pensado para facilitar el desarrollo nativo de aplicaciones iOS en iPhone o sistemas Apple. Es un lenguaje fuertemente tipado, aunque su declaración no siempre es necesaria gracias a su capacidad de inferir tipos.

Es un lenguaje que proporciona un manejo o control avanzado sobre el flujo de información y datos, así como la posibilidad de prever errores con bastante antelación y manipulación.

Algunas de las características de esta opción son:

- Código abierto.
- Comunidad más pequeña, debido a que es un lenguaje más reciente.
- Más difícil desarrollo y pruebas ya que exige tener un dispositivo IOS.
- Lenguaje orientado a objetos.



Ilustración 8. Swift

2.1.1.3. C# con Xamarin

Es un lenguaje de programación [15] multiparadigma desarrollado por Microsoft, que evoluciona de la familia de lenguajes C, tomando lo mejor de los lenguajes C y C++ . Forma parte de la plataforma .NET de Microsoft, un framework que se ha convertido en una de las principales plataformas de desarrollo debido a la facilidad que ofrece para la construcción de todo tipo de aplicaciones multiplataforma sólidas y duraderas.

Características de C#:

- Lenguaje orientado a objetos, polivalente y fácil de entender.
- Cierta familiaridad con él, ya que se ha usado en alguna asignatura del grado.

Xamarin [16] es una plataforma de código abierto para compilar aplicaciones modernas y con mejor rendimiento para iOS, Android y Windows con .NET. Su principal ventaja, es por tanto, que permite escribir aplicaciones multiplataforma, permitiendo que la aplicación se ejecute en una mayor variedad de dispositivos.

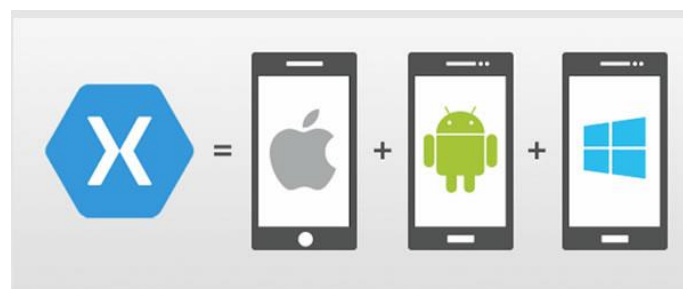


Ilustración 9. Xamarin

2.1.2. Base de datos

2.1.2.1. Base de datos relacional

Una base de datos relacional [17] es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí. Se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas. En una base de datos relacional, cada fila en una tabla es un registro, con una ID única. Las columnas de la tabla contienen los atributos de los datos y cada registro suele tener un valor para cada atributo, lo que simplifica la creación de relaciones entre los puntos de datos.

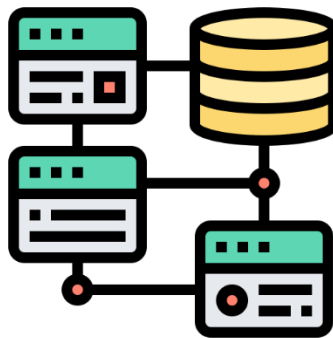


Ilustración 10. Base de datos relacional

A continuación, se barajan diferentes opciones de bases de datos relacionales

2.1.2.1.1. SQLite

Es una herramienta de software libre [18], que permite almacenar información en dispositivos que no necesitan tener mucha capacidad de hardware de una forma sencilla, eficaz, potente, rápida.

Agrega extensiones que facilitan su uso en cualquier ambiente de desarrollo. Esto permite, que SQLite soporte desde las consultas más básicas hasta las más complejas, y lo más importantes que se puede usar tanto en dispositivos móviles como en sistemas de escritorio, sin necesidad de realizar procesos complejos de importación y exportación de datos.

2.1.2.1.2. MySQL

MySQL [19] es un sistema open source de administración de bases de datos que es desarrollado y soportado por Oracle. Usa una estructura de cliente-servidor, en la que uno o más dispositivos (clientes) se conectan a un servidor a través de una red específica.

Es multihilo y multiusuario, y además cuenta con millones de instalaciones en todo el mundo. MySQL es el segundo motor de bases de datos relacional más utilizado en todo el mundo, por detrás de Oracle.

Además, es muy fácil de usar independientemente del lenguaje y el entorno, gratuita, rápida y al igual que SQLite tiene pocos requerimientos hardware y eficiencia de memoria, por todo esto, es una gran opción para desarrollar aplicaciones móviles.

2.1.2.2. Bases de datos documentales

Una base de datos documental [20] es una de las principales variantes de las bases de datos no relacionales. Se caracterizan por almacenar la información en registros, cada uno de los cuáles funciona como una unidad autónoma de información.

Como su propio nombre indica, las bases de datos orientadas a documentos utilizan documentos para el almacenamiento de todos los registros y los datos asociados a ellos. Cada uno de estos registros puede almacenar distintos tipos de datos. A su vez, los documentos que contienen los registros pueden tener diferentes formatos, desde archivos JSON o XML hasta documentos de texto.

Al ser un tipo de bases de datos no relacionales, otra de sus principales características es que la información no está contenida en tablas. Por el contrario, están pensadas para el almacenamiento de datos semiestructurados, los cuáles se organizan en documentos con valores asignados, los cuáles pueden ser consultados por los usuarios gracias a las diversas herramientas de consulta existentes.

Este tipo de base de datos, ofrecen una gran versatilidad, gracias a su estructura y flexibilidad.

2.1.2.2.1. MongoDB

Es un sistema de datos NOSQL, orientado a documentos de código abierto y escrito en C++. Al ser un proyecto de código abierto, sus binarios están disponibles para los sistemas operativos Windows, GNU/Linux, OS X y Solaris y es usado en múltiples proyectos o implementaciones.

La razón de esto es que MongoDB [21], al estar escrito en C++, cuenta con una más que notoria capacidad para aprovechar los recursos de la máquina, siendo su alta velocidad, su característica más importante.



Ilustración 11. MongoDB

2.1.2.2.2. Firebase

Firebase [22] de Google es una plataforma en la nube para el desarrollo de aplicaciones web y móvil. Está disponible para distintas plataformas (iOS, Android y web). Una de sus funcionalidades Firebase Firestore es una base de datos documental NoSQL caracterizada por su flexibilidad y escalabilidad en la nube y que permite almacenar y sincronizar los datos necesarios para el desarrollo del cliente y el servidor, así como dispositivos móviles y demás.

Al ser una base de datos en la nube, destaca su sincronización en tiempo real y sus triggers, eventos programables que pueden ocurrir si se captan cambios en la base de datos.



Ilustración 12. Firebase

2.2. PSI 7.2: SELECCIÓN DE LA ARQUITECTURA TECNOLÓGICA

Tras el análisis realizado previamente y teniendo en cuenta todas las opciones y la funcionalidad detallada en el PSI 1.2. En primer lugar, se decide desarrollar una aplicación móvil porque debido a la naturaleza del proyecto, es necesario que el usuario tenga completa libertad de movimiento para realizar ejercicios que posteriormente serán guardados

En cuanto a la elección de la tecnología, finalmente se decidió usar Android Studio con Java. El principal desencadenante es la familiaridad del equipo de desarrollo con el lenguaje de programación ya que en el Grado, se han impartido varias asignaturas usando ese lenguaje de programación.

Debido a la preferencia personal de Android respecto a IOS a nivel de sistema operativo, y la imposibilidad para acceder a un dispositivo móvil iPhone para realizar las pruebas, se descarta completamente la opción de IOS nativo con Swift. El alumno, al igual que su círculo cercano, cuenta con un terminal móvil Android. Desarrollar una aplicación para este sistema operativo, facilita mucho la implementación y ayuda enormemente a la hora de realizar las distintas pruebas.

C# con Xamarin, es una opción interesante por la compatibilidad con varios sistemas. Sin embargo, el desconocimiento total de Xamarin, la preferencia de Java respecto a C# y el deseo personal de aprender a usar mejor el entorno de desarrollo Android Studio, fomentaron que se desechara esa opción.

Como se ha dicho anteriormente, el alumno tiene un escaso conocimiento de Android Studio, pero tiene interés en aprender a usar esta herramienta debido a su popularidad y versatilidad a la hora de desarrollar diferentes proyectos.

En el caso de la base de datos, tras barajar las dos opciones, se decidió que la manera más cómoda y eficiente de crear la persistencia en la aplicación es mediante una base de datos documental. Tras un estudio de los datos que se desean persistir (pasos, calorías, rutas formadas por varios puntos geográficos), se deliberó que la forma más natural de hacerlo era a través de documentos, guardando para cada usuario los datos de cada día.

Además, como se desea que la aplicación este sincronizada con varios dispositivos, la solución que mejor se adapta es Firebase. El alumno tiene algo de experiencia con esta herramienta, adquirida toda ella en las prácticas del Grado. Además, le consta que es un instrumento que se



integra muy bien con aplicaciones móviles y que es muy popular. Por tanto, le interesa aprender a usarla con mayor soltura.

Se aprovecha este apartado para completar la selección de la API que se utilizará en el proyecto para el muestreo de mapas y localización del usuario. Tras realizar un estudio y la creación de varios proyectos de prueba, la API que ha sido más fácil de usar y que más opciones se consiguieron explotar fue la de MapBox. Para la toma de esta decisión también fue determinante que el plan de pagos de Google Maps puede desencadenar más problemas al empezar a cobrar cuando se gastan los 200\$. En cambio, Mapbox, al ofrecernos 25.000 clientes gratis (número que no se espera alcanzar), presenta una opción más interesante. A su vez, esta opción hace más sencillo realizar un seguimiento de gastos, lo que evita consumir capital no deseado.





CAPÍTULO 3: PLANIFICACIÓN DEL SISTEMA DE INFORMACIÓN

PLANIFICACIÓN

FASE DE

PSI

3.1. PLANIFICACIÓN DEL PROYECTO

En el siguiente apartado se va a detallar la planificación del proyecto, las actividades a realizar, la organización de los responsables, los resultados esperados y el plan de trabajo del proyecto. Seguidamente se elaborará un presupuesto inicial y un análisis de riesgos.

3.1.1. Identificación de Interesados

La identificación de interesados se realizó en [PSI 2.2: Organización del PSI](#).

3.1.2. OBS Y PBS

A continuación, se van a detallar el OBS y el PBS, de los que posteriormente se obtendrá el WBS.

3.1.1.1 OBS

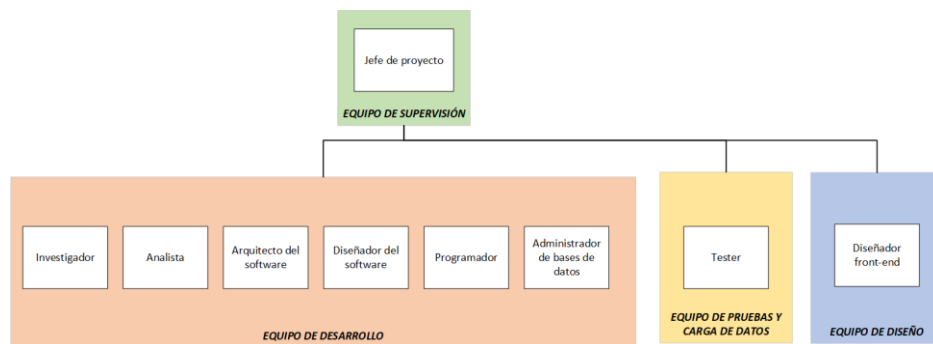


Ilustración 13. OBS

3.1.1.2 PBS

En el PBS se representa en forma de árbol los productos que han de obtenerse para completar el proyecto de manera satisfactoria.

Para facilitar su legibilidad, se ha dividido en varias imágenes:

3.1.1.2.1 PBS estructura general

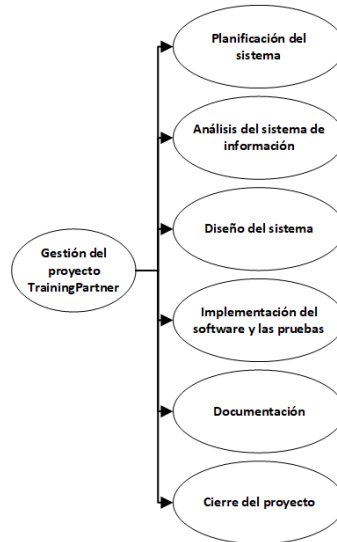


Ilustración 14. PBS Estructura general.

3.1.1.2.2 PBS Planificación del sistema

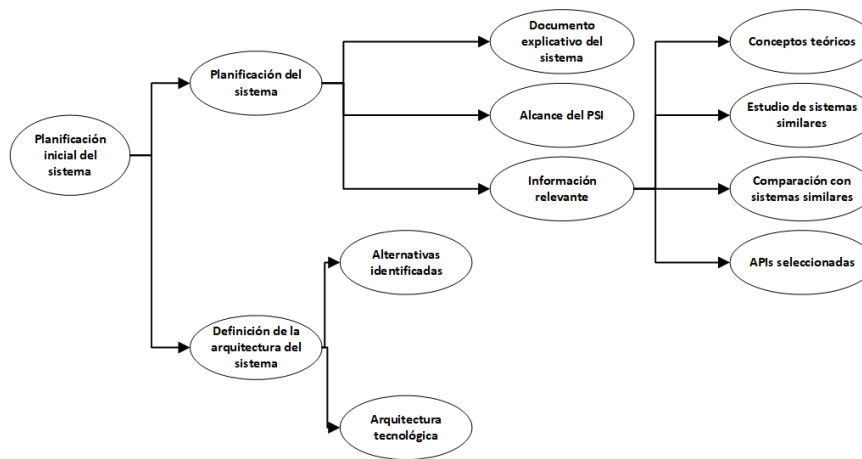


Ilustración 15. PBS Planificación del sistema.

3.1.1.2.3 PBS Análisis del sistema de información

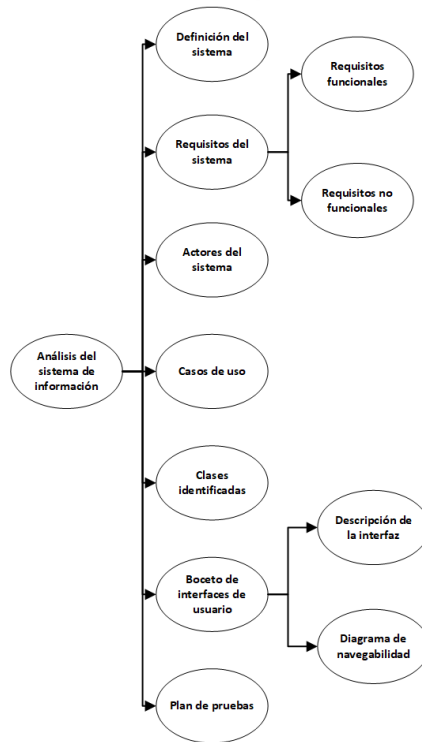


Ilustración 16. PBS Análisis.

3.1.1.2.4 PBS Diseño del sistema.

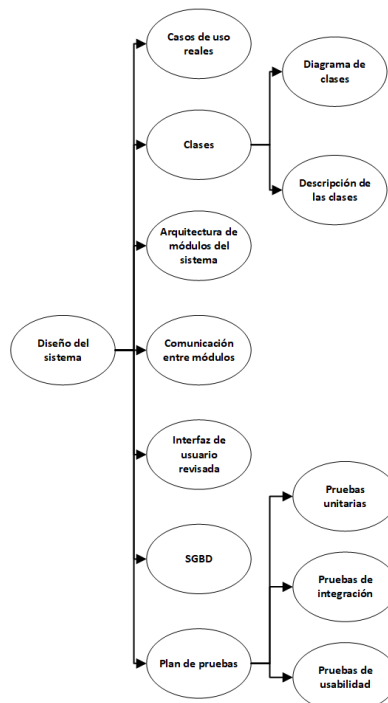


Ilustración 17. PBS Diseño.

3.1.1.2.5 PBS Desarrollo de software y pruebas

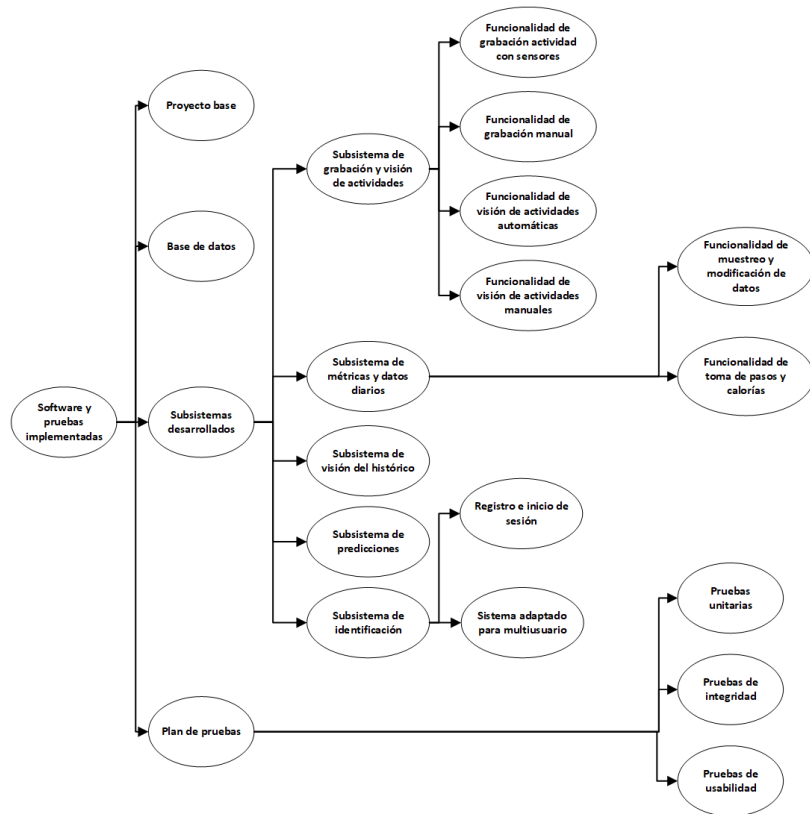


Ilustración 18. PBS Desarrollo.

3.1.1.2.6 PBS documentación

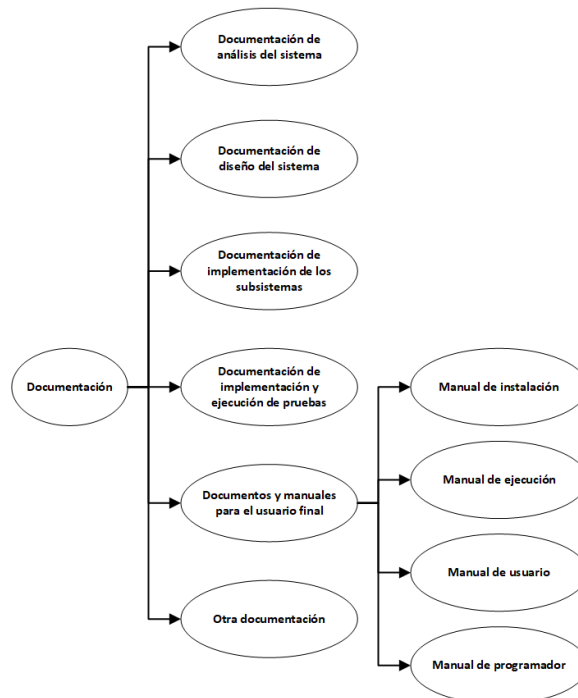


Ilustración 19. PBS Documentación.

3.1.1.2.7 Cierre del proyecto

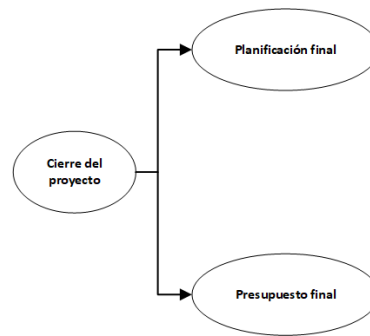


Ilustración 20. PBS Cierre del proyecto.

3.1.3. WBS y planificación inicial.

En este apartado se mostrará el WBS y posteriormente su traducción en la planificación inicial del proyecto.

3.1.1.3 WBS

El WBS indica las actividades a realizar para obtener los productos del PBS.

También se mostrará en diferentes secciones para facilitar su lectura.

3.1.1.3.1 Estructura general

Estas serán las fases principales en las que se realizará el proyecto:

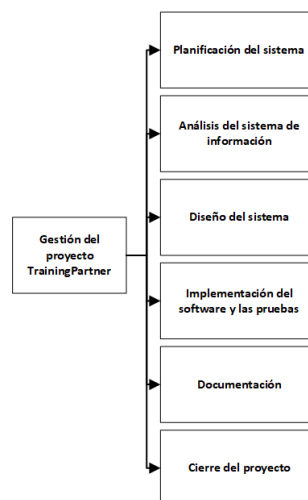


Ilustración 21. WBS Estructura general.

3.1.1.3.2 Planificación inicial del sistema

Esta primera fase es en la que se planifica el sistema y se define la arquitectura de este. Se explica el sistema que se va a realizar y se hace una búsqueda de APIs, soluciones similares y conceptos teóricos.

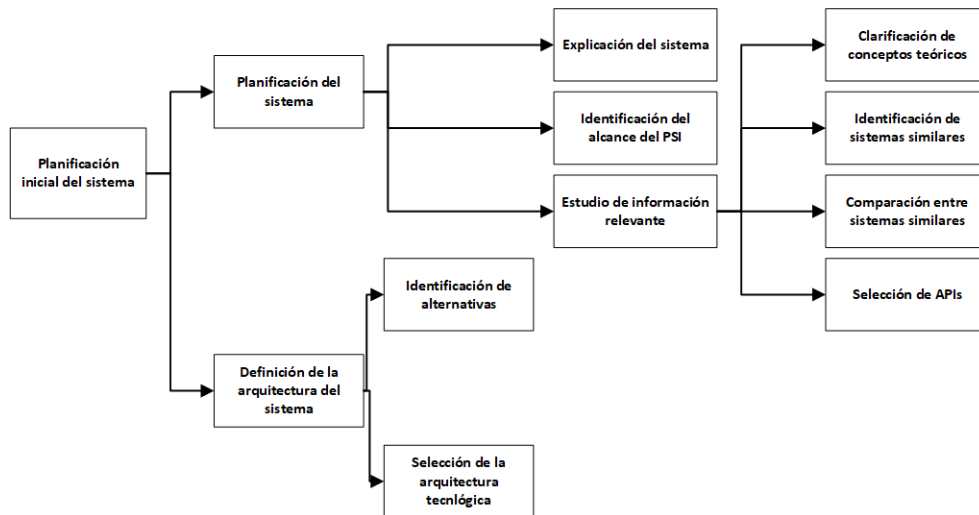


Ilustración 22. WBS Planificación del sistema.

3.1.1.3.3 Análisis del sistema de información

En esta fase se analiza todo el sistema a realizar, se detectan los requisitos, las clases, los casos de uso, se diseñan las pantallas, etc.

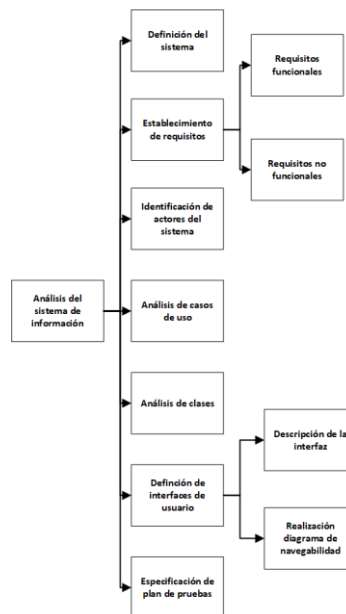


Ilustración 23. WBS Análisis.

3.1.1.3.4 Diseño del sistema

En esta fase se diseñarán todos los elementos detectados hasta ese momento, desde los casos de uso y las clases a las pruebas.

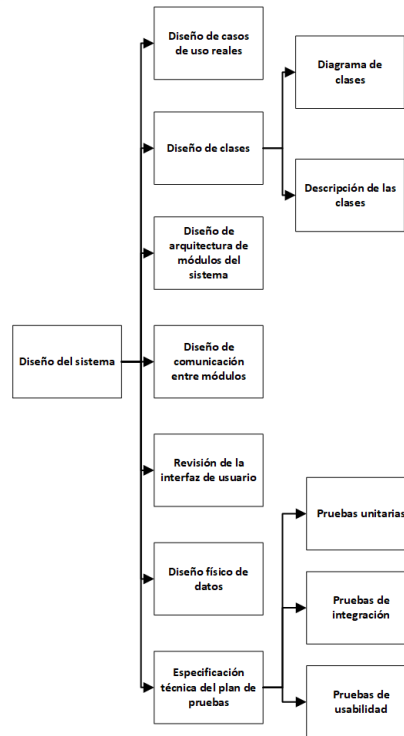


Ilustración 24. WBS Diseño.

3.1.1.3.5 Implementación del software y las pruebas

Es la fase en la que se desarrolla todo el software de la aplicación, se utiliza para elaborar los módulos del sistema y las pruebas de este.

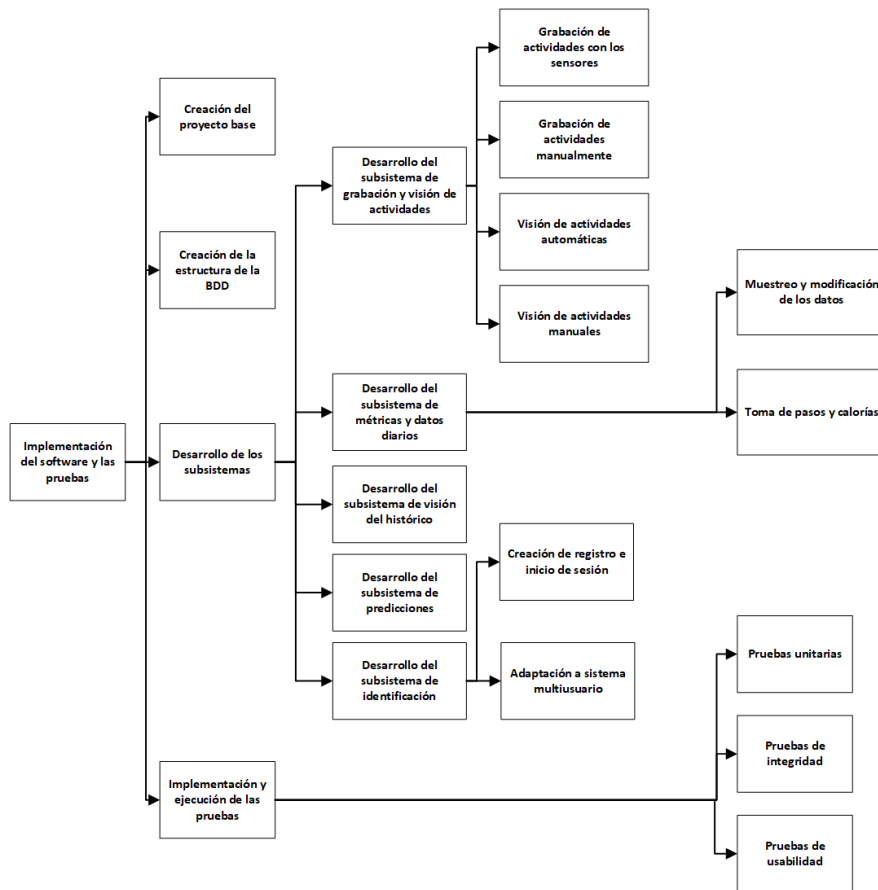


Ilustración 25. WBS Implementación.

3.1.1.3.6 Documentación

Esta fase es en la que se documentará todo lo realizado.

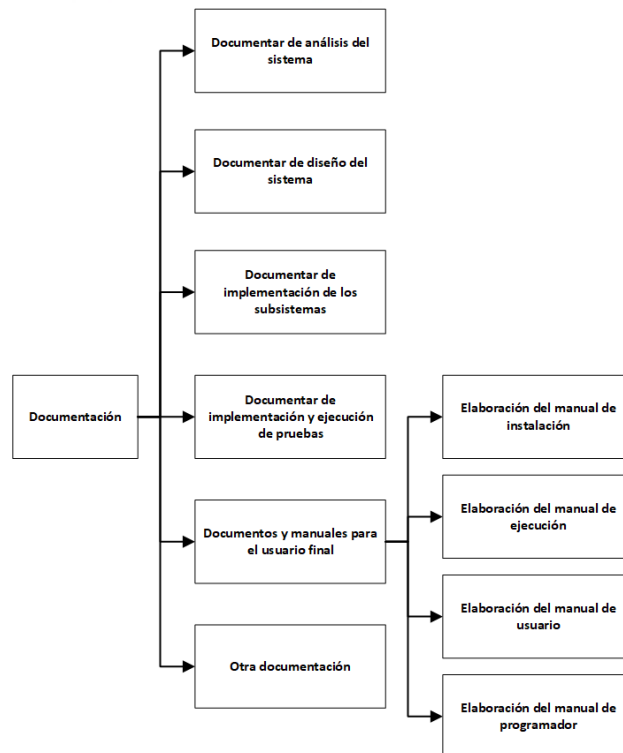


Ilustración 26. WBS Documentación.

3.1.1.3.7 Cierre del proyecto

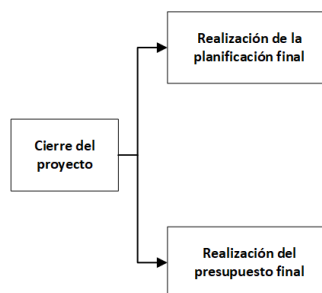


Ilustración 27. WBS Cierre del proyecto0.

3.1.1.4 Planificación inicial

En esta sección, a partir del WBS se genera la planificación inicial del sistema utilizando la herramienta Project [23]. El día de inicio del proyecto es el 19 de diciembre y según las estimaciones se terminaría el 24 de marzo.

Obviamente esta es una estimación, ya que se están teniendo en cuenta 4 días de trabajo semanales con 8 horas de trabajo al día. Habrá semanas que el trabajo sea mayor, alguna en la que sea menor y otras en las que no se le puedan dedicar horas al proyecto debido a otras obligaciones. Probablemente la fecha de fin se retrase varios días o semanas.

La planificación se ha dividido en las siguientes fases:

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1	Gestión del sistema	553 horas	lun 19/12/22	vie 24/03/23
1.1	Planificación del sistema	22 horas	lun 19/12/22	mié 21/12/22
1.2	Análisis del sistema de información	46 horas	mié 21/12/22	jue 29/12/22
1.3	Diseño del sistema	40 horas	jue 29/12/22	jue 05/01/23
1.4	Implementación del software y las pruebas	332 horas	jue 05/01/23	vie 03/03/23
1.5	Documentación	111 horas	lun 06/03/23	jue 23/03/23
1.6	Cierre del proyecto	2 horas	jue 23/03/23	vie 24/03/23

Tabla 3. Planificación general.

Se ha planeado que cada fase ocurra inmediatamente justo después de la anterior. En el caso de la documentación, aparece en el Project [23] justo después de la implementación y las pruebas, pero se irá documentando durante toda la vida útil del proyecto.

3.1.1.4.1 Fase de planificación del sistema.

La primera fase del proyecto es la que se corresponde con la planificación del sistema. Esta fase es en la que se realizarán las primeras reuniones con el tutor, se clarificará el sistema a implementar y el alcance de este. También se realizará el estudio de información relevante y la definición de la arquitectura del sistema. Finalmente se presupuestará todo el proyecto a partir de esta planificación.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.1	Planificación del sistema	22 horas	lun 19/12/22	mié 21/12/22
1.1.1	Planificación del sistema	16 horas	lun 19/12/22	mar 20/12/22
1.1.1.1	Explicación del sistema	2 horas	lun 19/12/22	lun 19/12/22
1.1.1.2	Identificación del alcance del PSI	4 horas	lun 19/12/22	lun 19/12/22
1.1.1.3	Estudio de información relevante	10 horas	lun 19/12/22	mar 20/12/22
1.1.1.3.1	Clarificación de conceptos teóricos	3 horas	lun 19/12/22	mar 20/12/22
1.1.1.3.2	Identificación de sistemas similares	1 hora	mar 20/12/22	mar 20/12/22
1.1.1.3.3	Comparación entre sistemas similares	2 horas	mar 20/12/22	mar 20/12/22
1.1.1.3.4	Selección de APIs	4 horas	mar 20/12/22	mar 20/12/22
1.1.2	Definición de la arquitectura del sistema	6 horas	mié 21/12/22	mié 21/12/22
1.1.2.1	Identificación de alternativas	3 horas	mié 21/12/22	mié 21/12/22
1.1.2.2	Selección de la arquitectura tecnológica	3 horas	mié 21/12/22	mié 21/12/22
1.1.3	Presupuesto inicial	8 horas	mié 21/12/22	jue 22/12/22

Tabla 4. Planificación inicial de la fase de planificación.

3.1.1.4.2 Fase de análisis

En la fase de análisis se definirá el sistema de manera formal, se identificarán los requisitos, tanto funcionales como no funcionales y se identificarán los actores del sistema, los casos de usos, las clases, la interfaz de usuario, la navegabilidad y las pruebas.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.2	Análisis del sistema de información	46 horas	jue 22/12/22	vie 30/12/22
1.2.1	Definición del sistema	3 horas	jue 22/12/22	vie 23/12/22
1.2.2	Establecimiento de requisitos	9 horas	vie 23/12/22	lun 26/12/22
1.2.2.1	Requisitos funcionales	7 horas	vie 23/12/22	vie 23/12/22
1.2.2.2	Requisitos no funcionales	2 horas	lun 26/12/22	lun 26/12/22
1.2.3	Identificación de actores del sistema	3 horas	lun 26/12/22	lun 26/12/22
1.2.4	Especificación de casos de uso	5 horas	lun 26/12/22	mar 27/12/22
1.2.5	Identificación de subsistemas de análisis	2 horas	mar 27/12/22	mar 27/12/22
1.2.6	Análisis de casos de uso	7 horas	mar 27/12/22	mié 28/12/22
1.2.7	Análisis de clases	8 horas	mié 28/12/22	jue 29/12/22
1.2.8	Definición de interfaces de usuario	9 horas	jue 29/12/22	vie 30/12/22
1.2.8.1	Descripción de la interfaz	5 horas	jue 29/12/22	jue 29/12/22
1.2.8.2	Realización de un diagrama de navegabilidad	2 horas	vie 30/12/22	vie 30/12/22
1.2.8.3	Especificación de plan de pruebas	2 horas	vie 30/12/22	vie 30/12/22

Tabla 5. Planificación inicial de la fase de análisis.

3.1.1.4.3 Fase de diseño

En esta fase se realizará todo el diseño de las fases analizadas anteriormente. Se diseñarán casos de uso reales, con todas las opciones posibles y las clases del sistema (describiendo sus métodos y atributos). También se utilizará esta fase para diseñar la arquitectura de los módulos y las comunicaciones entre estos. Finalmente se especificarán las pruebas a realizar.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.3	Diseño del sistema	40 horas	vie 30/12/22	vie 06/01/23
1.3.1	Diseño de casos de uso reales	7 horas	vie 30/12/22	lun 02/01/23
1.3.2	Diseño de clases	9 horas	lun 02/01/23	mar 03/01/23
1.3.2.1	Diagrama de clases	5 horas	lun 02/01/23	lun 02/01/23
1.3.2.2	Descripción de las clases	4 horas	mar 03/01/23	mar 03/01/23
1.3.3	Diseño de arquitectura de módulos del sistema	3 horas	mar 03/01/23	mar 03/01/23
1.3.4	Diseño de comunicación entre módulos	2 horas	mar 03/01/23	mié 04/01/23
1.3.5	Revisión de la interfaz de usuario	3 horas	mié 04/01/23	mié 04/01/23
1.3.6	Diseño físico de datos	3 horas	mié 04/01/23	mié 04/01/23
1.3.7	Especificación técnica del plan de pruebas	13 horas	mié 04/01/23	vie 06/01/23
1.3.7.1	Pruebas unitarias	5 horas	mié 04/01/23	jue 05/01/23



1.3.7.2	Pruebas de integración	5 horas	jue 05/01/23	vie 06/01/23
1.3.7.3	Pruebas de usabilidad	3 horas	vie 06/01/23	vie 06/01/23

Tabla 6. Planificación inicial de la fase de diseño.

3.1.1.4.4 Fase de desarrollo

En esta fase se implementará todo el software del sistema. Se comenzará creando el proyecto base y la base de datos, posteriormente se desarrollarán todos los subsistemas del proyecto. Se ha dividido la implementación en subsistemas para facilitar su priorización y desarrollo. Esta fase terminará implementando y ejecutando las pruebas diseñadas en la fase anterior. Algunos tests unitarios se ejecutarán y crearán durante el desarrollo de la aplicación, pero se han planificado de esta manera para facilitar su representación y estimación.

Debido a su complejidad, esta es la fase con mayor duración.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.4	Implementación del software y las pruebas	332 horas	vie 06/01/23	lun 06/03/23
1.4.1	Creación del proyecto base	1 día	vie 06/01/23	lun 09/01/23
1.4.2	Creación de la estructura de la base de datos	4 horas	lun 09/01/23	lun 09/01/23
1.4.3	Desarrollo de los subsistemas	256 horas	mar 10/01/23	mié 22/02/23
1.4.3.1	Desarrollo del subsistema de grabación y visión de actividades	104 horas	mar 10/01/23	jue 26/01/23
1.4.3.1.1	Grabación de actividades con los sensores	5 días	mar 10/01/23	lun 16/01/23
1.4.3.1.2	Grabación de actividades manualmente	2 días	mar 17/01/23	mié 18/01/23
1.4.3.1.3	Visión de actividades automáticas	4 días	jue 19/01/23	mar 24/01/23
1.4.3.1.4	Visión de actividades manuales	2 días	mié 25/01/23	jue 26/01/23
1.4.3.2	Desarrollo del subsistema de métricas y datos diarios	64 horas	vie 27/01/23	mar 07/02/23
1.4.3.2.1	Muestreo y modificación de los datos	4 días	vie 27/01/23	mié 01/02/23
1.4.3.2.2	Toma de pasos y calorías	4 días	jue 02/02/23	mar 07/02/23
1.4.3.3	Desarrollo del subsistema de visión del histórico	3 días	mié 08/02/23	vie 10/02/23
1.4.3.4	Desarrollo del subsistema de predicciones	3 días	lun 13/02/23	mié 15/02/23
1.4.3.5	Desarrollo del subsistema de identificación	40 horas	jue 16/02/23	mié 22/02/23
1.4.3.5.1	Creación de registro e inicio de sesión	3 días	jue 16/02/23	lun 20/02/23
1.4.3.5.2	Adaptación a sistema multiusuario	2 días	mar 21/02/23	mié 22/02/23
1.4.4	Implementación y ejecución de las pruebas	64 horas	jue 23/02/23	lun 06/03/23



1.4.4.1	Pruebas unitarias	3 días	jue 23/02/23	lun 27/02/23
1.4.4.2	Pruebas de integración	3 días	mar 28/02/23	jue 02/03/23
1.4.4.3	Pruebas de usabilidad	2 días	vie 03/03/23	lun 06/03/23

Tabla 7. Planificación inicial de la fase de implementación.

3.1.1.4.5 Fase de documentación

Como se ha señalado anteriormente, se ha programado esta fase al final para facilitar su seguimiento, pero el desarrollo de la documentación se llevará a cabo durante toda la vida útil del proyecto.

En esta fase se documentarán las anteriores y se crearán los manuales de instalación, de usuario y de ejecución.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.5	Documentación	111 horas	mar 07/03/23	vie 24/03/23
1.5.1	Documentar análisis del sistema	20 horas	mar 07/03/23	jue 09/03/23
1.5.2	Documentar diseño del sistema	20 horas	jue 09/03/23	lun 13/03/23
1.5.3	Documentar implementación de los subsistemas	20 horas	mar 14/03/23	jue 16/03/23
1.5.4	Documentar implementación y ejecución de pruebas	25 horas	jue 16/03/23	mar 21/03/23
1.5.5	Documentos y manuales para el usuario final	16 horas	mar 21/03/23	jue 23/03/23
1.5.5.1	Elaboración del manual de instalación	4 horas	mar 21/03/23	mié 22/03/23
1.5.5.2	Elaboración del manual de ejecución	4 horas	mié 22/03/23	mié 22/03/23
1.5.5.3	Elaboración del manual de usuario	8 horas	mié 22/03/23	jue 23/03/23
1.5.6	Otra documentación	10 horas	jue 23/03/23	vie 24/03/23

Tabla 8. Planificación inicial de la fase de documentación.

3.1.1.4.6 Fase de cierre del proyecto

Finalmente, se ha programado una fase para realizar el cierre del proyecto, en la que simplemente se realizará el presupuesto y planificación final.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.6	Cierre del proyecto	2 horas	vie 24/03/23	lun 27/03/23
1.6.1	Realización de la planificación final	2 horas	vie 24/03/23	lun 27/03/23
1.6.2	Realización del presupuesto final	2 horas	vie 24/03/23	lun 27/03/23

Tabla 9. Planificación inicial de la fase de cierre del proyecto.

3.1.1.5 Diagrama de Gantt

A continuación, se mostrarán los diagramas de Gantt para cada una de las fases detalladas en el apartado anterior.

3.1.1.5.1 General

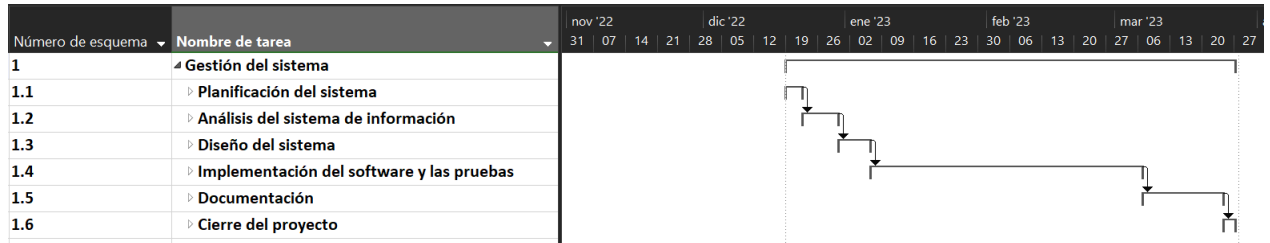


Ilustración 28. Diagrama de Gantt General.

3.1.1.5.2 Fase de planificación

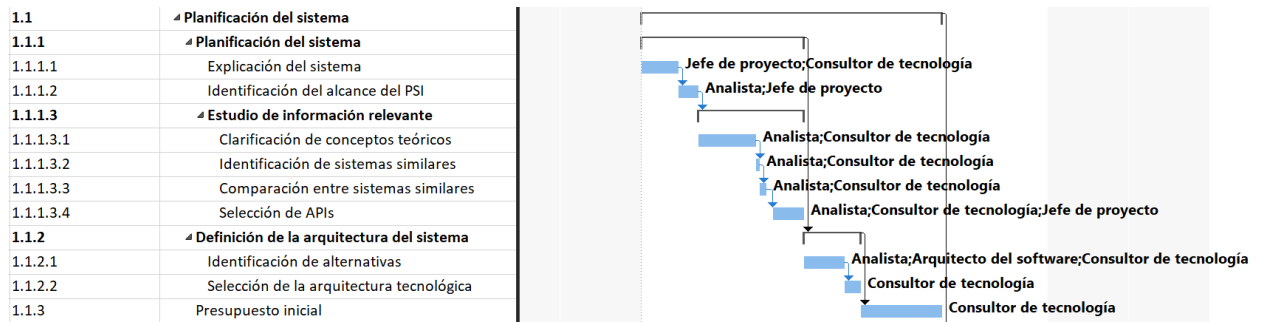


Ilustración 29. Diagrama de Gantt planificación.

3.1.1.5.3 Fase de análisis

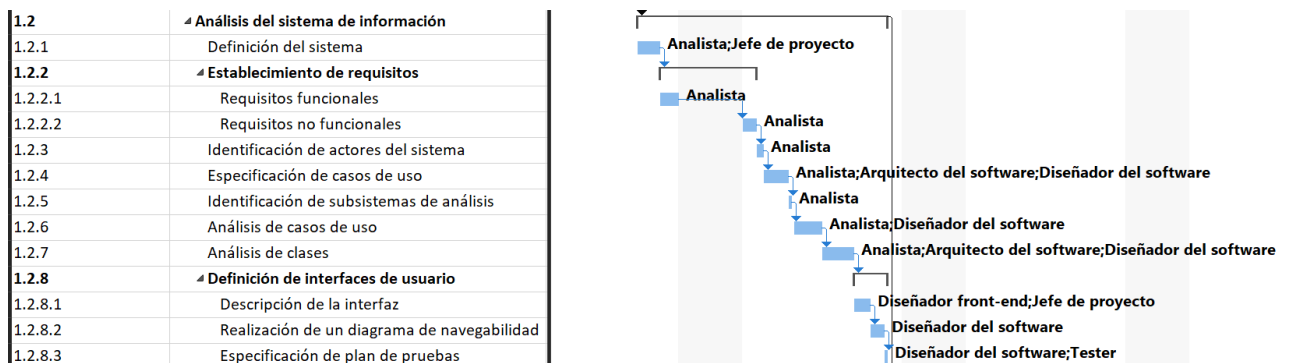


Ilustración 30. Diagrama de Gantt de análisis.

3.1.1.5.4 Fase de diseño

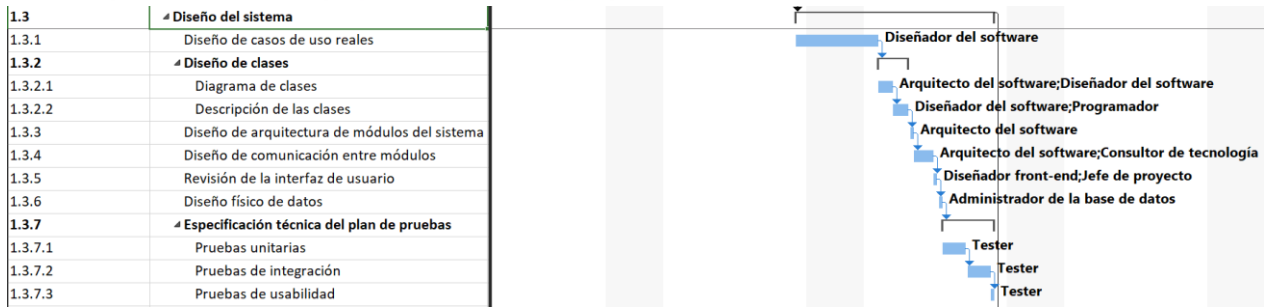


Ilustración 31. Diagrama de Gantt de diseño.

3.1.1.5.5 Fase de desarrollo

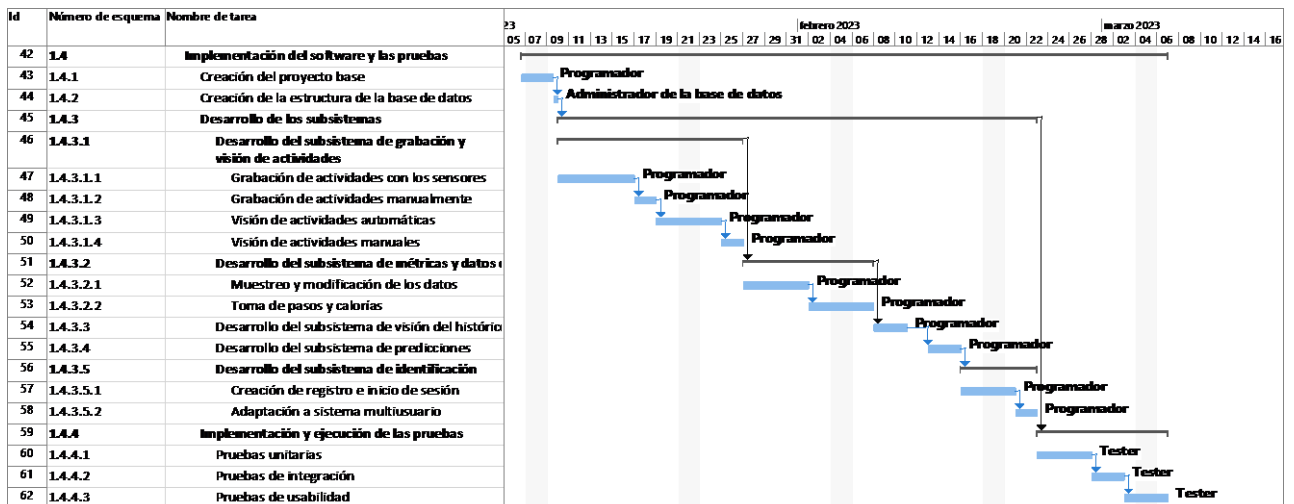


Ilustración 32. Diagrama de Gantt de desarrollo.

3.1.1.5.6 Fase de documentación

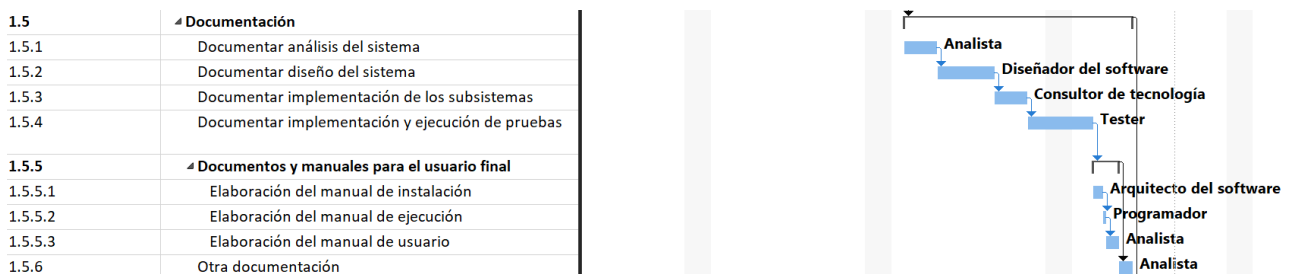


Ilustración 33. Diagrama de Gantt de documentación.

3.1.1.5.7 Fase de cierre de proyecto

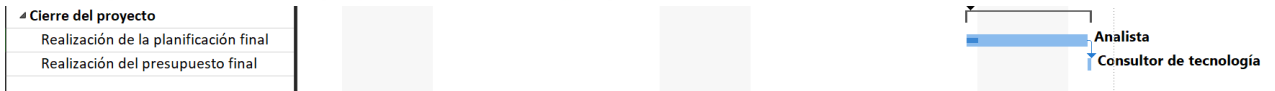


Ilustración 34. Diagrama de Gantt de cierre.

3.1.4. Riesgos

La planificación puede sufrir retrasos o adelantos debido a varios riesgos que pueden surgir durante el desarrollo del proyecto. Para poder mantener estos riesgos bajo control es preciso listarlos y crear un plan de gestión de estos.

3.1.1.6 Plan de gestión de riesgos

El plan de gestión de riesgos sirve para detallar la manera controlar y actuar contra los riesgos. Este plan aparecerá detallado en la sección [7.1. Plan de gestión de riesgos](#), dónde se explicará cómo gestionar cada uno de los riesgos identificados.

3.1.1.7 Identificación de riesgos

A fin de poder controlar los riesgos del sistema, en este apartado se crea una lista de riesgos potenciales. La lista está priorizada, es decir esta ordenada de mayor a menor impacto, en caso de coincidir el impacto, se ordenan por probabilidad.

Nombre del riesgo	Breve descripción	Categoría	Probabilidad	Impacto
Poca experiencia el desarrollo de aplicaciones móviles.	El alumno ha cursado únicamente una asignatura referente a dispositivos móviles y de una manera muy teórica. La falta de conocimientos puede retrasar el proyecto.	Técnico – Complejidad.	Muy Alta.	0,81
Poca experiencia con el desarrollo de pruebas.	El alumno nunca ha trabajado con las pruebas de espresso. Además, testear funcionalidades como la de grabación automática puede ser complejo. Esto puede retrasar la fecha de finalización del proyecto.	Técnico – Complejidad.	Muy Alta.	0,50
Falta de tiempo por el trabajo.	A la vez que se desarrolla el proyecto, el alumno está trabajando en una empresa a media jornada. La falta de tiempo puede retrasar el proyecto.	Técnico/Externo.	Alta.	0,39
No tener muy clara la funcionalidad y requisitos deseados.	El tema de este proyecto ha sido propuesto por el alumno. Si al inicio no se tiene muy clara la funcionalidad ni los requisitos, se puede perder tiempo de trabajo.	Requisitos.	Alta	0,39



Pérdida o mal funcionamiento del dispositivo.	El alumno sólo cuenta con un dispositivo Android en el que realizar las pruebas y ejecutar la aplicación. Dicho dispositivo es bastante antiguo y puede estropearse, lo cual desencadenaría un gasto económico y un retraso temporal.	Recursos.	Media.	0,28
Cambio en la política de pagos de alguna herramienta.	En el proyecto se utilizan dos herramientas de pago por uso (Google Cloud [24] y Mapbox [9]). Si las políticas de pago cambian, es posible que haya que modificar las herramientas, el uso que se haga de ellas o asumir un gasto económico que encarecería el presupuesto.	Recursos-Organizacional.	Baja.	0,27
Suspensión de alguna asignatura.	Un suspenso en alguna asignatura no permitiría al alumno entregar el proyecto en la fecha deseada.	Organizacional.	Baja.	0,17
Mala climatología.	Al vivir en Asturias es probable que la climatología dificulte salir a probar la aplicación.	Externo.	Muy Alta.	0,14
Lesión deportiva.	Una lesión deportiva además de influir negativamente en el estado de ánimo del alumno puede dificultarle mucho el desarrollo de pruebas de la aplicación.	Externo.	Baja.	0,09
Baja de miembro del proyecto.	Este riesgo se da si el alumno o el profesor (jefe de proyecto) se ponen de baja, lo cual retrasaría o paralizaría el desarrollo del proyecto.	Externo-Recurso.	Muy Baja.	0,06

Tabla 10. Riesgos identificados.

3.1.5. Presupuesto inicial

A partir de la planificación inicial se hará un presupuesto del coste de realización del proyecto. Al igual que la planificación, es una mera estimación que no será igual al que se obtendrá al finalizar el proyecto.

Para realizar el presupuesto se han utilizado los salarios que se utilizaron en el trabajo realizado en la asignatura *Dirección y planificación de proyectos informáticos* del grado, obtenidos del sitio web Glasdoor [25].

Personal	Precio/h
Analista	19,18 €
Arquitecto del software	30,69 €
Diseñador del software	24,30 €



Programador	24,30 €
Administrador de base de datos	24,30 €
Tester	24,30 €
Diseñador front-end	26,22 €
Jefe de proyecto	31,33 €
Consultor de tecnología	27,12 €

Tabla 11. Precio/h del personal.

3.1.1.8 Presupuesto de costes

El presupuesto de costes calculará el coste de realización del proyecto. En este caso, se dividirá en 7 partidas, una por cada una de las fases de la planificación y otra adicional para otros costes.

En este apartado se detallarán cada una de las partidas:



3.1.1.8.1 Partida 1: Planificación

I 1	I 2	I 3	I 4	Descripción	Cantidad	Unidades	Precio	Subtotal (2)	Subtotal (3)	Subtotal (4)	Total
001				Planificación del sistema							781,94 €
	001			Explicación del sistema						116,90 €	
			001	Consultor de tecnología	2	horas	27,12 €	54,24 €			
			002	Jefe de proyecto	2	horas	31,33 €	62,66 €			
	002			Identificación del alcance del PSI						202,04 €	
			001	Analista	4	horas	19,18 €	76,72 €			
			002	Jefe de proyecto	4	horas	31,33 €	125,32 €			
	003			Estudio de información relevante						463,00 €	
		001		Clarificación de conceptos teóricos					138,90 €		
			001	Analista	3	horas	19,18 €	57,54 €			
			002	Consultor de tecnología	3	horas	27,12 €	81,36 €			
		002		Identificación de sistemas similares					46,30 €		
			001	Analista	1	horas	19,18 €	19,18 €			
			002	Consultor de tecnología	1	horas	27,12 €	27,12 €			
		003		Comparación entre sistemas similares					92,60 €		
			001	Analista	2	horas	19,18 €	38,36 €			
			002	Consultor de tecnología	2	horas	27,12 €	54,24 €			
		004		Selección de APIs					185,20 €		
			001	Analista	4	horas	19,18 €	76,72 €			
			002	Consultor de tecnología	4	horas	27,12 €	108,48 €			
002				Definición de la arquitectura del sistema							312,33 €
	001			Identificación de alternativas						230,97 €	
			001	Analista	3	horas	19,18 €	57,54 €			
			002	Arquitecto del software	3	horas	30,69 €	92,07 €			



		003	Consultor de tecnología	3	horas	27,12 €	81,36 €		
002			Selección de la arquitectura tecnológica						81,36 €
		001	Consultor de tecnología	3	horas	27,12 €	81,36 €		
003			Presupuesto inicial						216,96 €
		001	Consultor de tecnología	8	horas	27,12 €	216,96 €		
TOTAL									1.311,23 €

Tabla 12. Partida de planificación.

3.1.1.8.2 Partida 2: Análisis

I 1	I 2	I 3	Descripción	Cantidad	Unidades	Precio	Subtotal (2)	Subtotal (3)	Total
001			Definición del sistema						151,53 €
		001	Analista	3	horas	19,18 €	57,54 €		
		002	Jefe de proyecto	3	horas	31,33 €	93,99 €		
002			Establecimiento de requisitos						172,62 €
	001		Requisitos funcionales					134,26 €	
		001	Analista	7	horas	19,18 €	134,26 €		
	002		Requisitos no funcionales					38,36 €	
		001	Analista	2	horas	19,18 €	38,36 €		
003			Identificación de actores del sistema						57,54 €
		001	Analista	3	horas	19,18 €	57,54 €		
004			Especificación de casos de uso						370,85 €
		001	Analista	5	horas	19,18 €	95,90 €		
		002	Arquitecto del software	5	horas	30,69 €	153,45 €		
		003	Diseñador del software	5	horas	24,30 €	121,50 €		
005			Identificación de subsistemas de análisis						38,36 €



	001	Analista	2 horas	19,18 €	38,36 €		
006		Análisis de casos de uso					304,36 €
	001	Analista	7 horas	19,18 €	134,26 €		
	001	Diseñador del software	7 horas	24,30 €	170,10 €		
007		Análisis de clases					593,36 €
	001	Analista	8 horas	19,18 €	153,44 €		
	002	Arquitecto del software	8 horas	30,69 €	245,52 €		
	003	Diseñador del software	8 horas	24,30 €	194,40 €		
008		Definición de interfaces de usuario					384,95 €
	001	Descripción de la interfaz				287,75 €	
	001	Diseñador front-end	5 horas	26,22 €	131,10 €		
	002	Jefe de proyecto	5 horas	31,33 €	156,65 €		
	002	Realización de un diagrama de navegabilidad				48,60 €	
	001	Diseñador del software	2 horas	24,30 €	48,60 €		
	003	Especificación de plan de pruebas				48,60 €	
	001	Tester	2 horas	24,30 €	48,60 €		
TOTAL							2.073,57 €

Tabla 13. Partida de análisis.

3.1.1.8.3 Partida 3: Diseño

I 1	I 2	I 3	Descripción	Cantidad	Unidades	Precio	Subtotal (2)	Subtotal (3)	Total
001			Diseño de casos de uso reales						170,10 €
		001	Diseñador del software	7	horas	24,30 €	170,10 €		
002			Diseño de clases						372,15 €
	001		Diagrama de clases					274,95 €	



	001	Arquitecto del software	5	horas	30,69 €	153,45 €		
	002	Diseñador del software	5	horas	24,30 €	121,50 €		
	002	Desscripción de las clases					97,20 €	
	001	Diseñador del software	4	horas	24,30 €	97,20 €		
003		Diseño de arquitectura de módulos del sistema						92,07 €
	001	Arquitecto del software	3	horas	30,69 €	92,07 €		
004		Diseño de comunicación entre módulos						115,62 €
	001	Arquitecto del software	2	horas	30,69 €	61,38 €		
	002	Consultor de tecnología	2	horas	27,12 €	54,24 €		
005		Revisión de la interfaz de usuario						172,65 €
	001	Diseñador front-end	3	horas	26,22 €	78,66 €		
	002	Jefe de proyecto	3	horas	31,33 €	93,99 €		
006		Diseño físico de datos						72,90 €
	001	Administrador de base de datos	3	horas	24,30 €	72,90 €		
007		Especificación técnica del plan de pruebas						315,90 €
	001	Pruebas unitarias					121,50 €	
	001	Tester	5	horas	24,30 €	121,50 €		
	002	Pruebas de integración					121,50 €	
	001	Tester	5	horas	24,30 €	121,50 €		
	003	Pruebas de usabilidad					72,90 €	
	001	Tester	3	horas	24,30 €	72,90 €		
TOTAL								1.311,39 €

Tabla 14. Partida de diseño.

3.1.1.8.4 Partida 4: Implementación



I 1	I 2	I 3	I 4	Descripción	Cantidad	Unidades	Precio	Subtotal (2)	Subtotal (3)	Subtotal (4)	Total
001				Creación del proyecto base							194,40 €
			001	Programador	1	día	24,30 €	194,40 €			
002				Creación de la estructura de la base de datos							97,20 €
			001	Administrador de base de datos	4	horas	24,30 €	97,20 €			
003				Desarrollo de los subsistemas							6.220,80 €
	001			Desarrollo del subsistema de grabación y visión de actividades						2.527,20 €	
		001		Grabación de actividades con los sensores					972,00 €		
			001	Programador	5	día	24,30 €	972,00 €			
		002		Grabación de actividades manualmente					388,80 €		
			001	Programador	2	día	24,30 €	388,80 €			
		003		Visión de actividades automáticas					777,60 €		
			001	Programador	4	día	24,30 €	777,60 €			
		004		Visión de actividades manuales					388,80 €		
			001	Programador	2	día	24,30 €	388,80 €			
	002			Desarrollo del subsistema de métricas y datos diarios						1.555,20 €	
		001		Muestreo y modificación de los datos					777,60 €		
			001	Programador	4	día	24,30 €	777,60 €			
		002		Toma de pasos y calorías					777,60 €		
			001	Programador	4	día	24,30 €	777,60 €			
	003			Desarrollo del subsistema de visión del histórico						583,20 €	
			001	Programador	3	día	24,30 €	583,20 €			



004			Desarrollo del subsistema de predicciones						583,20 €	
		001	Programador	3	día	24,30 €	583,20 €			
005			Desarrollo del subsistema de identificación						972,00 €	
		001	Creación de registro e inicio de sesión						583,20 €	
		001	Programador	3	día	24,30 €	583,20 €			
		002	Adaptación a sistema multiusuario						388,80 €	
		001	Programador	2	día	24,30 €	388,80 €			
004			Implementación y ejecución de las pruebas							1.555,20 €
	001		Pruebas unitarias						583,20 €	
		001	Tester	3	día	24,30 €	583,20 €			
	002		Pruebas de integración						583,20 €	
		001	Tester	3	día	24,30 €	583,20 €			
	002		Pruebas de usabilidad						388,80 €	
		001	Tester	2	día	24,30 €	388,80 €			
TOTAL										8.067,60 €

Tabla 15. Partida de implementación.

3.1.1.8.5 Partida 5: Documentación

I 1	I 2	I 3	Descripción	Cantidad	Unidades	Precio	Subtotal (2)	Subtotal (3)	Total
001			Documentar análisis del sistema						383,60 €
		001	Analista	20	horas	19,18 €	383,60 €		
002			Documentar diseño del sistema						486,00 €
		001	Diseñador del software	20	horas	24,30 €	486,00 €		
003			Documentar implementación de los subsistemas						542,40 €



		001	Consultor de tecnología	20	horas	27,12 €	542,40 €		
004			Documentar implementación y ejecución de pruebas						607,50 €
		001	Tester	25	horas	24,30 €	607,50 €		
005			Documentos y manuales para el usuario final						373,40 €
	001		Elaboración del manual de instalación					122,76 €	
		001	Arquitecto del software	4	horas	30,69 €	122,76 €		
	002		Elaboración del manual de ejecución					97,20 €	
		001	Programador	4	horas	24,30 €	97,20 €		
	003		Elaboración del manual de usuario					153,44 €	
		001	Analista	8	horas	19,18 €	153,44 €		
006			Otra documentación						191,80 €
		001	Analista	10	horas	19,18 €	191,80 €		
TOTAL									2.584,70 €

Tabla 16. Partida de documentación

3.1.1.8.6 Partida 6: Cierre de proyecto

I 1	I 2	Descripción	Cantidad	Unidades	Precio	Subtotal (2)	Total
001		Realización de la planificación final					38,36 €
	001	Analista	2	horas	19,18 €	38,36 €	
002		Realización del presupuesto final					54,24 €
	001	Consultor de tecnología	2	horas	27,12 €	54,24 €	
TOTAL							92,60 €



Tabla 17. Partida de cierre del proyecto.

3.1.1.8.7 Partida 7: Otros gastos

I1	Ítem	Consumo(hora)	Unidades	Horas	Precio (€) de KW/h	Total
001	Consumo eléctrico	0,22	kW/h	563	0,18 €	22,29 €
TOTAL						22,29 €

Tabla 18. Partida de otros gastos.

3.1.1.8.8 Resumen del presupuesto de costes

A continuación, se muestra el resumen del presupuesto de costes, que asciende a **15.463,38€**.

Como se ha señalado anteriormente esta es únicamente una estimación del presupuesto a partir de la planificación inicial, los valores finales pueden tener cambios significativos debidos a retrasos, malas estimaciones, riesgos o problemas.



Código	Partida	Importe
01	Planificación del sistema	1.311,23 €
02	Análisis del sistema de información	2.073,57 €
03	Diseño del sistema	1.311,39 €
04	Implementación del software y las pruebas	8.067,60 €
05	Documentación	2.584,70 €
06	Cierre del proyecto	92,60 €
07	Otros gastos	22,29 €
TOTAL		15.463,38 €

Tabla 19. Resumen del presupuesto de costes.

3.1.1.9 Presupuesto de Cliente

Tras realizar el presupuesto general, se crea un presupuesto para el cliente, en el cual no se deben de percibir ciertas partidas como la de planificación, cierre del proyecto y otros gastos. Además, en este presupuesto se deben de tener en cuenta un porcentaje de beneficio, que se ha estimado que sea del 25% basándose en el tiempo invertido y la dificultad del proyecto.

Para cuadrar los presupuestos, el beneficio y las partidas que no se van a mostrar aparecerán ponderadas en las que sí que percibirá el cliente.

Beneficio	25%	3.865,85 €
Partidas no visibles		1.426,12 €
Total a ponderar		5.291,97 €
Porcentaje de ponderación		38%

Tabla 20. Datos para el presupuesto de Cliente.

Como se puede ver en la tabla superior, se calcula un porcentaje de ponderación del 38%, que se obtiene dividiendo el total a ponderar entre la suma de los precios de las partidas que si se van a mostrar al cliente.

Por tanto, el presupuesto que percibe el cliente es:

Presupuesto de Cliente		
Código	Partida	Importe
01	Análisis del sistema de información	2.855,29 €
02	Diseño del sistema	1.805,78 €
03	Implementación del software y las pruebas	11.109,04 €
04	Documentación	3.559,12 €
TOTAL		19.329,23 €

Tabla 21. Presupuesto de Cliente.

El presupuesto del cliente es de **19.329,23 €**, obteniendo así un beneficio de **3.865,85 €**. Es importante destacar, que todos estos presupuestos y beneficios son calculados sin tener en cuenta el IVA.

3.2. EJECUCIÓN DEL PROYECTO

3.2.1. Bitácora de incidencias

Durante el proyecto hubo una serie de causas que hicieron que la fecha de finalización del proyecto se alejara de la planificada. Se planeaba tardar 425 horas menos en realizar el proyecto. Es importante destacar que este tiempo de retraso no ha sido tiempo trabajado en su totalidad, ya que durante la realización del proyecto ha habido semanas en las que no se ha podido decir nada de tiempo a este por diversas razones.

De las 425 horas, sólo 62 son de trabajo extra (que se han planificado erróneamente). El resto de tiempo ha sido por semanas en las que ha habido obligaciones de exámenes, trabajo u ocio. Por tanto, el presupuesto sólo se verá modificado por las 62 horas extra.

A continuación, se muestra la bitácora de incidencias.

ID	Tipo de incidencia	Fecha de creación	Fecha de cierre	Descripción de la incidencia	Efecto en la planificación
01	Retraso	22/12/2022	03/01/2023	Las festividades de navidad causaron que se parara el desarrollo del proyecto.	Se comienza el análisis del sistema de la información 11 días más tarde de lo planeado.
02	Fallo de planificación.	03/01/2023	04/01/2023	Al no tener aún claros los requisitos del sistema, se tarda más tiempo en elaborar los requisitos funcionales.	Requisitos funcionales -> 3 horas más de lo planeado.
03	Fallo de planificación.	06/01/2023	09/01/2023	La falta de dominio sobre la herramienta de creación de diagramas Visio [26], hace que se alargue el análisis de casos de uso.	Análisis de casos de uso -> 3 horas más de lo esperado.
04	Retraso.	18/01/2023	15/02/2023	El comienzo del segundo cuatrimestre y el inicio en un nuevo puesto de trabajo, hacen que la elaboración del proyecto se pause casi un mes.	Se comienza la implementación del software y las pruebas con casi un mes de retraso respecto a lo planeado.

05	Fallo de planificación.	16/02/2023	07/04/2023	Los subsistemas de grabación de actividades automáticas y de toma de pasos y calorías requieren un tiempo de estudio ya que no se dominan los servicios en segundo plano.	Grabación de actividades con los sensores -> 2 días más de lo planeado (16 horas). Toma de pasos y calorías -> 2 días más de lo planeado (16 horas).
06	Retraso.	07/04/2023	28/04/2023	La cercanía de los exámenes finales del grado y el desconocimiento de la realización de pruebas, hacen que el comienzo de estas se retrase.	El comienzo de la implementación y ejecución de las pruebas se retrasa 21 días
07	Fallo de planificación.	28/04/2023	04/05/2023	La inexperiencia con las pruebas hace que las unitarias tarden más tiempo en desarrollarse.	Desarrollo de las pruebas unitarias -> 2 días más de lo planeado
08	Fallo de planificación.	12/05/2023	29/05/2023	Fallo en la planificación de la documentación.	Se tarda 8 horas más de lo previsto en realizar la documentación del sistema.

Tabla 22. Bitácora.

Sumando todos los fallos de planificación, se obtienen las 62 horas extra, mientras que la suma de todas las incidencias es igual a las 425 horas de descuadre.

3.2.2. Riesgos

En este apartado se realizará el estudio de algunos de los riesgos del sistema.

Riesgo 1	Poca experiencia en el desarrollo de aplicaciones móviles.
Descripción	Se había planificado tiempo extra para paliar este riesgo, pero aún así las dificultades con los servicios en segundo plano alargaron el tiempo de desarrollo del proyecto.
Retraso	4 días.
Evento de bitácora	Evento 05.

Tabla 23. Riesgo: Poca experiencia en el desarrollo de aplicaciones móviles.

Riesgo 4	No tener muy clara la funcionalidad y requisitos deseados.
Descripción	Se disminuyeron los efectos de este riesgo realizando la planificación antes que el análisis, pero aun así, afectó negativamente al proyecto alargando el proceso de elaboración de requisitos funcionales.
Retraso	3 horas.
Evento de bitácora	Evento 02.

Tabla 24. Riesgo: No tener muy clara la funcionalidad del sistema.

Riesgo 7	Suspense de alguna asignatura.
Descripción	Se llevaron al día las materias cursadas, y el riesgo no afectó al proyecto.
Retraso	Ninguno.
Evento de bitácora	Ninguno.

Tabla 25. Suspense de alguna asignatura.

Riesgo 3	Falta de tiempo por el trabajo.
Descripción	Los primeros meses de trabajo tomaron más tiempo del planificado y eso creó retrasos en el comienzo de fases del proyecto.
Retraso	30 días.
Evento de bitácora	Evento 04.

Tabla 26. Riesgo: Falta de tiempo por el trabajo.

Riesgo 8	Mala climatología.
Descripción	La climatología no fue un impedimento en esta ocasión para realizar entrenamientos y pruebas, ya que hubo gran sequía en toda España en los meses de marzo y abril. Además, se utilizaron aplicaciones que falsean la ubicación para probar el sistema.
Retraso	Ninguno.
Evento de bitácora	Ninguno.

Tabla 27. Riesgo: Mala climatología.

Riesgo 2	Poca experiencia con el desarrollo de pruebas.
-----------------	---



Descripción	Aunque se había tenido en cuenta este riesgo a la hora de realizar la planificación, afectó igualmente al proyecto retrasándolo un tiempo.
Retraso	2 días.
Evento de bitácora	Evento 07.

Tabla 28. Riesgo: Poca experiencia con el desarrollo de pruebas.

3.3. CIERRE DEL PROYECTO

En el siguiente apartado se muestra la planificación y presupuestos finales. Al terminar se hará una comparación entre la planificación y presupuestos iniciales y los que se llevaron a cabo.

3.3.1. Planificación final

Como se mencionó en el apartado de Bitácora, el proyecto ha sufrido una serie de retrasos, causados por un fallo de la planificación inicial o por parones debidos a otras obligaciones. En general, aunque hubo semanas que no se cumplieron, las jornadas de 8 horas pudieron realizarse, ya que se llegó a un acuerdo con la empresa en la que trabajaba el alumno para tener los viernes libres.

A continuación, se muestra la planificación final, cuyo seguimiento se ha realizado a través de la herramienta Project [23].

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1	Gestión del sistema	988 horas	lun 19/12/22	jue 08/06/23
1.1	Planificación del sistema	30 horas	lun 19/12/22	jue 22/12/22
1.2	Análisis del sistema de información	52 horas	mar 03/01/23	mié 11/01/23
1.3	Diseño del sistema	40 horas	mié 11/01/23	mié 18/01/23
1.4	Implementación del software y las pruebas	652 horas	mié 15/02/23	jue 08/06/23
1.5	Documentación	119 horas	vie 12/05/23	jue 01/06/23
1.6	Cierre del proyecto	4 horas	jue 01/06/23	vie 02/06/23

Tabla 29. Planificación final general.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.1	Planificación del sistema	30 horas	lun 19/12/22	jue 22/12/22
1.1.1	Planificación del sistema	16 horas	lun 19/12/22	mar 20/12/22
1.1.1.1	Explicación del sistema	2 horas	lun 19/12/22	lun 19/12/22
1.1.1.2	Identificación del alcance del PSI	4 horas	lun 19/12/22	lun 19/12/22
1.1.1.3	Estudio de información relevante	10 horas	lun 19/12/22	mar 20/12/22
1.1.1.3.1	Clarificación de conceptos teóricos	3 horas	lun 19/12/22	mar 20/12/22
1.1.1.3.2	Identificación de sistemas similares	1 hora	mar 20/12/22	mar 20/12/22
1.1.1.3.3	Comparación entre sistemas similares	2 horas	mar 20/12/22	mar 20/12/22
1.1.1.3.4	Selección de APIs	4 horas	mar 20/12/22	mar 20/12/22
1.1.2	Definición de la arquitectura del sistema	6 horas	mié 21/12/22	mié 21/12/22

1.1.2.1	Identificación de alternativas	3 horas	mié 21/12/22	mié 21/12/22
1.1.2.2	Selección de la arquitectura tecnológica	3 horas	mié 21/12/22	mié 21/12/22
1.1.3	Presupuesto inicial	8 horas	mié 21/12/22	jue 22/12/22

Tabla 30. Planificación final de planificación del sistema.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.2	Análisis del sistema de información	52 horas	mar 03/01/23	mié 11/01/23
1.2.1	Definición del sistema	3 horas	mar 03/01/23	mar 03/01/23
1.2.2	Establecimiento de requisitos	12 horas	mar 03/01/23	mié 04/01/23
1.2.2.1	Requisitos funcionales	10 horas	mar 03/01/23	mié 04/01/23
1.2.2.2	Requisitos no funcionales	2 horas	mié 04/01/23	mié 04/01/23
1.2.3	Identificación de actores del sistema	3 horas	mié 04/01/23	jue 05/01/23
1.2.4	Especificación de casos de uso	5 horas	jue 05/01/23	jue 05/01/23
1.2.5	Identificación de subsistemas de análisis	2 horas	jue 05/01/23	vie 06/01/23
1.2.6	Análisis de casos de uso	10 horas	vie 06/01/23	lun 09/01/23
1.2.7	Análisis de clases	8 horas	lun 09/01/23	mar 10/01/23
1.2.8	Definición de interfaces de usuario	9 horas	mar 10/01/23	mié 11/01/23
1.2.8.1	Descripción de la interfaz	5 horas	mar 10/01/23	mar 10/01/23
1.2.8.2	Realización de un diagrama de navegabilidad	2 horas	mié 11/01/23	mié 11/01/23
1.2.8.3	Especificación de plan de pruebas	2 horas	mié 11/01/23	mié 11/01/23

Tabla 31. Planificación final de análisis del sistema.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.3	Diseño del sistema	40 horas	mié 11/01/23	mié 18/01/23
1.3.1	Diseño de casos de uso reales	7 horas	mié 11/01/23	jue 12/01/23
1.3.2	Diseño de clases	9 horas	jue 12/01/23	vie 13/01/23
1.3.2.1	Diagrama de clases	5 horas	jue 12/01/23	jue 12/01/23
1.3.2.2	Descripción de las clases	4 horas	vie 13/01/23	vie 13/01/23
1.3.3	Diseño de arquitectura de módulos del sistema	3 horas	vie 13/01/23	vie 13/01/23
1.3.4	Diseño de comunicación entre módulos	2 horas	vie 13/01/23	lun 16/01/23
1.3.5	Revisión de la interfaz de usuario	3 horas	lun 16/01/23	lun 16/01/23
1.3.6	Diseño físico de datos	3 horas	lun 16/01/23	lun 16/01/23
1.3.7	Especificación técnica del plan de pruebas	13 horas	lun 16/01/23	mié 18/01/23
1.3.7.1	Pruebas unitarias	5 horas	lun 16/01/23	mar 17/01/23
1.3.7.2	Pruebas de integración	5 horas	mar 17/01/23	mié 18/01/23
1.3.7.3	Pruebas de usabilidad	3 horas	mié 18/01/23	mié 18/01/23

Tabla 32. Planificación final de diseño del sistema.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.4	Implementación del software y las pruebas	652 horas	mié 15/02/23	jue 08/06/23
1.4.1	Creación del proyecto base	1 día	mié 15/02/23	mié 15/02/23
1.4.2	Creación de la estructura de la base de datos	4 horas	jue 16/02/23	jue 16/02/23
1.4.3	Desarrollo de los subsistemas	288 horas	jue 16/02/23	vie 07/04/23
1.4.3.1	Desarrollo del subsistema de grabación y visión de actividades	120 horas	jue 16/02/23	jue 09/03/23
1.4.3.1.1	Grabación de actividades con los sensores	7 días	jue 16/02/23	lun 27/02/23
1.4.3.1.2	Grabación de actividades manualmente	2 días	lun 27/02/23	mié 01/03/23
1.4.3.1.3	Visión de actividades automáticas	4 días	mié 01/03/23	mar 07/03/23
1.4.3.1.4	Visión de actividades manuales	2 días	mar 07/03/23	jue 09/03/23
1.4.3.2	Desarrollo del subsistema de métricas y datos diarios	80 horas	jue 09/03/23	jue 23/03/23
1.4.3.2.1	Muestreo y modificación de los datos	4 días	jue 09/03/23	mié 15/03/23
1.4.3.2.2	Toma de pasos y calorías	6 días	mié 15/03/23	jue 23/03/23
1.4.3.3	Desarrollo del subsistema de visión del histórico	3 días	jue 23/03/23	mar 28/03/23
1.4.3.4	Desarrollo del subsistema de predicciones	3 días	mar 28/03/23	vie 31/03/23
1.4.3.5	Desarrollo del subsistema de identificación	40 horas	vie 31/03/23	vie 07/04/23
1.4.3.5.1	Creación de registro e inicio de sesión	3 días	vie 31/03/23	mié 05/04/23
1.4.3.5.2	Adaptación a sistema multiusuario	2 días	mié 05/04/23	vie 07/04/23
1.4.4	Implementación y ejecución de las pruebas	80 horas	vie 28/04/23	jue 11/05/23
1.4.4.1	Pruebas unitarias	5 días	vie 28/04/23	jue 04/05/23
1.4.4.2	Pruebas de integración	3 días	vie 05/05/23	mar 09/05/23
1.4.4.3	Pruebas de usabilidad	2 días	mié 10/05/23	jue 11/05/23

Tabla 33. Planificación final de implementación.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.5	Documentación	119 horas	vie 12/05/23	jue 01/06/23
1.5.1	Documentar análisis del sistema	22 horas	vie 12/05/23	mar 16/05/23
1.5.2	Documentar diseño del sistema	23 horas	mar 16/05/23	vie 19/05/23

1.5.3	Documentar implementación de los subsistemas	23 horas	vie 19/05/23	mié 24/05/23
1.5.4	Documentar implementación y ejecución de pruebas	25 horas	mié 24/05/23	lun 29/05/23
1.5.5	Documentos y manuales para el usuario final	16 horas	lun 29/05/23	mié 31/05/23
1.5.5.1	Elaboración del manual de instalación	4 horas	lun 29/05/23	mar 30/05/23
1.5.5.2	Elaboración del manual de ejecución	4 horas	mar 30/05/23	mar 30/05/23
1.5.5.3	Elaboración del manual de usuario	8 horas	mar 30/05/23	mié 31/05/23
1.5.6	Otra documentación	10 horas	mié 31/05/23	jue 01/06/23

Tabla 34. Planificación final de documentación.

Número de esquema	Nombre de tarea	Duración	Comienzo	Fin
1.6	Cierre del proyecto	4 horas	jue 01/06/23	vie 02/06/23
1.6.1	Realización de la planificación final	2 horas	jue 01/06/23	vie 02/06/23
1.6.2	Realización del presupuesto final	2 horas	vie 02/06/23	vie 02/06/23

Tabla 35. Planificación final de cierre del proyecto.

3.3.2. Presupuesto Final de Costes

En el siguiente apartado se muestra el resumen del presupuesto final de costes teniendo en cuenta los cambios hechos en la planificación. Para obtener más detalle se puede acceder a los datos adjuntos de PlanificacionFinal-UO277955 y PresupuestoFinal-UO277955.

Finalmente, también se mostrará un resumen del presupuesto percibido por el cliente.

Código	Partida	Importe
01	Planificación del sistema	1.311,23 €
02	Análisis del sistema de información	2.261,55 €
03	Diseño del sistema	1.311,39 €
04	Implementación del software y las pruebas	9.234,00 €
05	Documentación	2.777,32 €
06	Cierre del proyecto	92,60 €
07	Otros gastos	24,75 €
TOTAL		17.012,84 €

Tabla 36. Presupuesto final de costes.

Beneficio	25%	4.253,21 €
Partidas no visibles		1.428,58 €
Total a ponderar		5.681,79 €
Porcentaje de ponderación		36%

Tabla 37. Datos para el cálculo de presupuesto final de Cliente.

Presupuesto de Cliente		
Código	Partida	Importe
01	Análisis del sistema de información	3.086,08 €
02	Diseño del sistema	1.789,50 €
03	Implementación del software y las pruebas	12.600,58 €
04	Documentación	3.789,89 €
TOTAL		21.266,05 €

Tabla 38. Presupuesto final de Cliente.

3.3.3. Comparativa de planificación y presupuesto inicial y final

Como se ha comentado anteriormente, inicialmente el proyecto estaba planeado para tener una duración de 563 horas y terminar el 27 de marzo. Debido a los retrasos, el proyecto finalmente ha contado con 988 horas (de las cuales 625 han sido de trabajo) y se ha desarrollado desde el 19 de diciembre de 2022 al 2 de junio de 2023.

A continuación, se muestra una comparación de los costes de las partidas:

Código	Partida	Importe planificación inicial	Importe planificación final	Incremento de la partida final respecto a la inicial.
01	Planificación del sistema	1.311,23 €	1.311,23 €	- €
02	Análisis del sistema de información	2.073,57 €	2.261,55 €	187,98 €
03	Diseño del sistema	1.311,39 €	1.311,39 €	- €
04	Implementación del software y las pruebas	8.067,60 €	9.234,00 €	1.166,40 €
05	Documentación	2.584,70 €	2.777,32 €	192,62 €
06	Cierre del proyecto	92,60 €	92,60 €	- €
07	Otros gastos	22,29 €	24,75 €	2,46 €
TOTAL		15.463,38 €	17.012,84 €	
DIFERENCIA				1.549,46 €

Tabla 39. Comparativa de presupuesto final e inicial.

Como se puede comprobar, el proyecto costaría **1.549,46€** más de lo esperado.

3.3.4. Informe de lecciones aprendidas

La planificación de un sistema es una parte que cuenta con gran importancia dentro del desarrollo de este. Aunque a menudo no se tenga tan en cuenta, permite organizar los esfuerzos para llevar a cabo un proyecto de manera ordenada. Esto se traduce en un desarrollo más rápido y eficaz.

Además, una buena planificación permite ajustar mejor los gastos, lo cual en ciertos proyectos con presupuesto limitado es indispensable. Por ejemplo, en este trabajo una planificación que no hubiese tenido en cuenta los riesgos hubiese aumentado aún más el gasto total.



También se ha aprendido que por muchos esfuerzos que se hagan, es muy difícil estimar las planificaciones y presupuestos, ya que hay muchos factores externos que hacen que los tiempos y plazos varíen.





CAPÍTULO 4: ANÁLISIS DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

ASI



4.1. DEFINICION DEL SISTEMA

4.1.1. Determinación del Alcance del sistema

El sistema consistirá en el prototipo de una aplicación móvil cuyo objetivo es recopilar datos y métricas de la actividad física de los usuarios a través de los sensores del teléfono. Se pretende realizar una aplicación sencilla, para poder llegar a la mayor variedad de público posible. Se desarrollarán 3 funciones principales: toma de datos diarios del usuario, grabación de entrenamientos y predicción de tiempos en distintas distancias.

Primeramente, la aplicación necesitará del registro del usuario, usando un nombre y una contraseña. De esta manera, se podrá acceder a los datos independientemente del dispositivo en el que se use la aplicación.

La aplicación medirá de manera automática (siempre que se esté ejecutando en primer o segundo plano), los pasos que da y las calorías que quema el usuario en el día a día. Estos datos, se tomarán a través de diferentes algoritmos y de los sensores del dispositivo móvil.

En cuanto a los entrenamientos, el usuario podrá iniciar un entrenamiento en cualquier momento que lo desee. Cuando se inicie una actividad, mediante el GPS del dispositivo, se comenzará a guardar la ruta que sigue el usuario, la distancia que lleva recorrida, la velocidad a la que lo está haciendo y el tiempo que ha pasado desde que comenzó el entrenamiento. La actividad se podrá finalizar, parar y reanudar en cualquier momento.

Los entrenamientos también podrán guardarse de manera manual, el usuario introducirá la distancia recorrida y el tiempo que ha durado la actividad, y esta se guardará de la misma manera que las que fueron registradas haciendo uso de los sensores del dispositivo.

Una vez finalizada la actividad, se podrá ver toda la información resumida: La distancia recorrida, la velocidad media a la que se desplazó el usuario y el tiempo total de duración del entrenamiento. En el caso de las actividades que fueron registradas mediante los sensores del dispositivo, también se mostrará la velocidad media de cada kilómetro y un mapa con el recorrido realizado. A cualquier tipo de actividad finalizada se le podrá adjuntar una imagen que se mostrará junto al resto de datos de esta. Si lo desea, el usuario podrá eliminar las actividades una vez hayan sido finalizadas.

Los entrenamientos podrán ser consultados junto a los pasos y calorías de ese día. Toda esta información, será guardada en un histórico. Se podrá acceder a los registros de días pasados mediante un calendario.

Se crearán predicciones de tiempo que, a partir de entrenamientos pasados, harán uso de algoritmos para aproximar el tiempo que le puede tomar al usuario realizar las siguientes distancias específicas: 1km, 10km, 21km y 42 km (maratón). Esta funcionalidad solo estará disponible si se han registrado al menos 3 entrenamientos de una distancia mínima de 3 km. Las predicciones irán cambiando a medida que se guarden nuevas actividades.

4.2. ASI 2: ESTABLECIMIENTO DE REQUISITOS

4.2.1. Requisitos funcionales

4.2.1.1. Registro de usuario

RReg1. El sistema permitirá a los usuarios autenticarse en la aplicación

RReg1.1. El usuario deberá indicar los siguientes campos de manera obligatoria.

RReg1.1.1 Nombre usuario

RReg1.1.2 Contraseña

RReg1.1.3. Nombre

RReg1.1.4. Apellidos

RReg1.1.5. Peso en kg

RReg1.1.6. Altura en cm

RReg1.1.7. El sexo que podrá ser:

RReg1.1.7.1. Hombre

RReg1.1.7.2. Mujer

RReg1.1.8. Fecha de nacimiento

RReg1.2. El sistema comprobará que ninguno de los campos sea vacío.

RReg1.3. El sistema comprobará que la contraseña cumpla las siguientes condiciones para ser válida:

RReg1.3.1. Longitud de 8 caracteres.

RReg1.3.2. Contiene mayúsculas y minúsculas.

RReg1.3.3. Contiene al menos un carácter especial (@, #, _)

RReg1.3.4. En el caso que sea válida, el sistema:

RReg1.3.4.1 Comprobará que el nombre de usuario no exista en la base de datos.

RReg1.3.4.1.1 En el caso de que no exista, el sistema:

RReg1.3.4.1.1.1 Guardará los valores en la base de datos

RReg1.3.4.1.1.1.1 Nombre de usuario

RReg1.3.4.1.1.1.2 Contraseña encriptada.

RReg1.3.4.1.1.1.3. Nombre



RReg1.3.4.1.1.1.4. Apellidos

RReg1.3.4.1.1.1.5. Peso

RReg1.3.4.1.1.1.6. Altura

RReg1.3.4.1.1.1.7. Sexo

RReg1.3.1.1.1.1.8. Fecha de nacimiento

RReg1.2.4.1.1.2 Autenticará al usuario.

RReg1.2.4.1.1.3 El sistema guardará las credenciales, para que el usuario sólo tenga que identificarse una vez.

RReg1.2.4.1.2 En el caso de que exista:

RReg1.2.4.1.2.1. Se mostrará el error por pantalla

RReg1.2.4.1.2.2. El sistema pedirá que se vuelva a introducir el nombre de usuario.

RReg1.2.5 En el caso que sea inválida:

RReg1.2.5.1. Se mostrará un mensaje de error.

RReg1.2.5.2. El sistema pedirá que se vuelva a introducir la contraseña.

4.2.1.1. Inicio de sesión

Rlni1. El sistema permitirá a los usuarios registrados y no identificados iniciar sesión en la aplicación.

Rlni1.1. El sistema pedirá al usuario que se introduzcan los siguientes campos:

Rlni1.1.1. Nombre de usuario

Rlni1.1.2. Contraseña

Rlni1.2. El sistema comprobará que la combinación de Rlni1.1.1 y Rlni1.1.2 existe en la base de datos.

Rlni1.2.1. En el caso de que existan:

Rlni1.2.1.1. Se iniciará sesión

Rlni1.2.1.2. El sistema informará al usuario.

Rlni1.2.2. En el caso de que no existan:

Rlni1.2.2.1. El sistema mostrará el error.

Rlni1.2.2.2. Se pedirá que se vuelvan a introducir las credenciales.

Rlni1.3. El sistema guardará las credenciales, para que el usuario sólo tenga que identificarse una vez.



4.2.1.2. *Cerrar sesión*

RCer1. El sistema permitirá al usuario identificado cerrar la sesión.

RCer1.1. Si la sesión se cierra correctamente:

RCer1.1.1. El sistema eliminará las credenciales, para que no se inicie la sesión automáticamente.

RCer1.1.2. Se redirigirá al usuario a la ventana de inicio de sesión, para que pueda autenticarse.

RCer1.3. Si la sesión no se cierra correctamente, el sistema mostrará un mensaje de error al usuario.

4.2.1.3. *Grabar una actividad*

RGra1. El sistema permitirá al usuario grabar una actividad de caminata o running de manera automática mediante los sensores del dispositivo.

RGra1.1. El sistema, mediante el GPS del dispositivo accederá a la ubicación del usuario.

RGra1.1.1. Si la aplicación no tiene permisos para usar la ubicación, se pedirán.

RGra1.1.2. Si la ubicación del dispositivo no está activada, se avisará al usuario.

RGra1.1.3. Una vez establecida la localización, se permitirá iniciar la actividad.

RGra1.2. Al iniciar la actividad, el sistema:

RGra1.2.1. Guardará y mostrará la distancia en kilómetros que ha recorrido el usuario desde que se inició la actividad.

RGra1.2.2. Calculará y expondrá el tiempo que ha transcurrido desde que comenzó la actividad.

RGra1.2.3. Mostrará la velocidad a la que se está desplazando el usuario.

RGra1.2.4. Guardará la ubicación del usuario cada 100 metros.

RGra1.2.5. Mostrará el ritmo en min/km al que se está moviendo el usuario.

RGra1.3. El sistema permitirá pausar la grabación de la actividad en cualquier momento.

RGra1.3.1. El sistema dejará de realizar RGra1.2.1, RGra1.2.2, RGra1.2.3, RGra1.2.4 Y RGra1.2.5.

RGra1.3.2. El sistema permitirá al usuario reanudar la actividad.

RGra1.4. El sistema permitirá finalizar la actividad en cualquier momento.

RGra1.4.1. El sistema dejará de realizar RGra1.2.1, RGra1.2.2, RGra1.2.3 y RGra1.2.4.

RGra1.4.2. Se preguntará al usuario si quiere finalizar la actividad.

RGra1.4.3. En el caso de que confirme la finalización:



RGra1.4.3.1. El sistema guardará los siguientes campos en la base de datos:

RGra1.4.3.1.1. Distancia total en kilómetros.

RGra1.4.3.1.2. Tiempo de duración de la actividad.

RGra1.4.3.1.3. Tiempo que se tardó en hacer cada kilómetro.

RGra1.4.3.1.4. Algunos puntos geográficos del trayecto seguido, que permitirán trazar posteriormente la ruta

RGra1.4.3.1.5. Un marcador para reflejar que la actividad se grabó usando los sensores del teléfono.

RGra1.4.3.2. Se avisará al usuario:

RGra1.4.3.2.1. Si la actividad ha sido guardada correctamente.

RGra1.4.3.2.2. Si ha ocurrido algún error.

RGra1.4.3.3. El sistema redirigirá al usuario a la ventana de visualización de la actividad.

RGra1.4.4. En el caso de que no confirme la finalización de la actividad:

RGra1.4.4.1. Se reanudará RGra1.2.1, RGra1.2.2, RGra1.2.3 y RGra1.2.4.

RGra2. El sistema permitirá guardar entrenamientos de manera manual.

RGra2.1. El sistema pedirá los siguientes campos al usuario:

RGra2.1.1. Distancia en kilómetros realizada.

RGra2.1.2. Tiempo que ha tardado en realizar la actividad.

RGra2.2. El sistema calculará:

RGra2.2.1. La velocidad media a la que se realizó la actividad

RGra2.2.2. El ritmo medio al que se desplazó el usuario.

RGra2.3. El sistema guardará en la base de datos:

RGra2.3.1. RGra2.1.1.

RGra2.3.2. RGra2.1.2

RGra2.3.3. Un marcador para reflejar que se trata de una actividad grabada manualmente.

RGra2.4. Se avisará al usuario:

RGra2.5.1. Si la actividad ha sido guardada correctamente

RGra2.5.2. Si ha ocurrido algún error.

RGra2.5. El sistema redirigirá al usuario a la ventana de visualización de la actividad.

4.2.1.4. *Gestionar una actividad realizada*

RGes1. El sistema permitirá mostrar una actividad ya realizada.

RGes1.1. Si la actividad se grabó con los sensores del dispositivo:

RGes1.1.1. Se mostrarán los siguientes campos:

RGes1.1.1.2. Un mapa interactivo con la ruta que siguió el usuario en esa actividad.

RGes1.1.1.3. La distancia total en kilómetros que se recorrió en esa actividad.

RGes1.1.1.4. El tiempo que duró la actividad.

RGes1.1.1.5. La velocidad media a la que se desplazó el usuario.

RGes1.1.1.6. El tiempo que se tardó en realizar cada uno de los kilómetros.

RGes1.1.1.7. El ritmo medio (min/km) al que se desplazó el usuario

RGes1.2. Si la actividad se grabó de manera manual:

RGes1.2.1. Se mostrarán RGe1.1.1.3, RGe1.1.1.4 y RGe1.1.1.5 y RGe1.1.1.7.

RGes2. El sistema permitirá borrar una actividad ya realizada.

RGes2.1. A la hora de mostrar la actividad, se le dará la opción al usuario de eliminarla.

RGes2.1.1. El sistema preguntará al usuario si está seguro de borrarla.

RGes2.1.1.1. Si el usuario confirma:

RGes2.1.1.1.1. Se eliminará la actividad de la base de datos.

RGes2.1.1.1.2. Se avisará al usuario:

RGes2.1.1.1.2.1 Si ha ocurrido algún problema.

RGes2.1.1.1.2.2. Si se ha borrado correctamente.

RGes2.1.1.2. En caso negativo, se continuará mostrando la actividad.

4.2.1.5. *Visualización de métricas generales*

RVis1. El sistema, por medio del acelerómetro del dispositivo, calculará los pasos que realiza el usuario.

RVis1.1. Al final de cada día, se guardará en la base de datos el total de pasos realizados.



RVis2. El sistema, a partir de los datos de entrenamiento y los pasos diarios, hará una estimación de las calorías quemadas.

RVis2.1. Al final de cada día, se guardará en la base de datos el total de pasos realizados.

RVis3. El sistema mostrará al usuario los siguientes datos:

RVis3.1. Nombre

RVis3.2. Apellidos

RVis3.3. Peso.

RVis3.4. Altura.

RVis3.5. Una foto identificativa

RVis4. Se permitirá al usuario modificar:

RVis4.1. RVis3.3.

RVis4.1.1. El usuario introducirá el nuevo peso.

RVis4.1.1.1. Si el peso tiene un formato correcto:

RVis4.1.1.1.1 Se le preguntará al usuario si está seguro de cambiarlo.

RVis4.1.1.1.1.1. En caso afirmativo:

RVIS4.1.1.1.1.1.1. El peso se actualizará en la base de datos.

RVis4.1.1.1.1.2. En caso negativo:

RVIS4.1.1.1.1.2.1. No se cambiará el dato.

RVis4.1.1.2. Si el peso tiene un formato incorrecto:

RVis4.1.1.2.1. No se le permitirá al usuario modificar el valor.

RVis4.1.1.2.2. El sistema mostrará un mensaje explicativo.

RVis4.2. RVis3.4.

RVis4.2.1. El usuario introducirá la nueva altura.

RVis4.2.1.1. Si la altura tiene un formato correcto:

RVis4.2.1.1.1. Se le preguntará al usuario si esta seguro de hacer el cambio.

RVis4.2.1.1.1.1. En caso afirmativo:

RVis4.2.1.1.1.1.1. La altura se modificará en la base de datos.

RVis4.2.1.1.1.2. En caso negativo:

RVIS4.2.1.1.1.2.1. La altura no se modificará.



RVis4.2.1.2. Si el formato es incorrecto:

RVis4.2.1.2.1. No se le permitirá al usuario modificar el valor

RVis4.2.1.2.2. El sistema mostrará un mensaje explicativo.

RVis4.3. RVis3.5.

RVis4.3.1. El sistema pedirá al usuario permisos para:

RVis4.3.1.1. Acceder a la cámara.

RVis4.3.1.2. Acceder a los archivos.

RVis4.3.1.3. Si no se dan estos permisos:

RVis4.3.1.3.1 Se seguirá mostrando la ventana principal.

RVis4.3.2. El usuario podrá decidir si:

RVis4.3.2.1. Seleccionar una foto de la galería

RVis4.3.2.2. Hacer una foto.

RVis4.3.3. El usuario seleccionará la parte de la foto que le interesa.

RVis4.3.4. La foto será actualizada

RVis4.3.5. Se guardará la imagen en la base de datos.

4.2.1.6. *Predicciones de tiempos*

RPred1. El sistema, a partir de los entrenamientos guardados, predecirá el tiempo que realizará el usuario en las distancias de 1km, 10km, 21km y 42km.

RPred1.1. Para que se realice la predicción, el usuario debe haber guardado al menos 3 entrenamientos de una distancia mínima de 3km.

RPred1.1.1. En el caso de que no se cumpla la condición, en el espacio reservado para las predicciones, aparecerá un mensaje explicativo.

RPred1.2. Las predicciones irán cambiando a medida que se añadan nuevas actividades.

4.2.1.7. *Visualización de un histórico*

RHis1. El sistema permitirá visualizar un histórico de los pasos, calorías y actividades que se realizaron cada día.

RHis1.1. El usuario podrá interactuar con el histórico a través de un calendario.

RHis1.1.1. Si se accede a un día del que no se guardan registros:

RHis1.1.1.1. Se mostrará un mensaje explicativo.



RHis1.1.1.2. Se le ofrecerá al usuario la opción de volver a la ventana del calendario.

RHis1.1.2. Si se accede a un día del que se guardan registros:

RHis1.1.2.1. Se mostrará una ventana con los siguientes campos:

RHis1.1.2.1.1. Pasos

RHis1.1.2.1.2. Calorías.

RHis1.1.2.1.3. Todas las actividades que se realizaron ese día.

RHis1.1.2.1.3.1. Se podrá acceder a los detalles de esas actividades.

RHis1.1.2.2. Se le ofrecerá al usuario la opción de volver a la ventana del calendario.

RHis1.1.3. El usuario podrá filtrar el histórico por:

RHis1.1.3.1. Mes.

RHis1.1.3.2. Año.

4.2.2. Requisitos no funcionales

4.2.2.1. Tecnológicos

RNFTec1. El cálculo de los pasos y la grabación de entrenamientos, deben de poder ejecutarse en segundo plano y con la pantalla del dispositivo apagada.

RNFTec2. La aplicación debe de poder ejecutarse en cualquier versión de Android igual o posterior a la 8.0.

RNFTec3. La aplicación debe de conectarse con la base de datos en tiempo real Firebase Firestore, para guardar toda la información de las actividades, los pasos, las calorías y los datos relacionados con la identificación de usuarios.

RNFTec4. La aplicación debe de tener los siguientes permisos:

RNFTec4.1. Ubicación.

RNFTec4.2. Acceso a archivos.

RNFTec4.3. Ejecución en segundo plano.

4.2.2.2. Usabilidad.

RNFUsa1. La aplicación debe de poder ser usada por cualquier persona con un conocimiento reducido de dispositivos móviles.



RNFUsa2. El diseño debe de ser intuitivo y simple, ya que se persigue que este prototipo sea utilizado por cualquier tipo de usuario.

4.2.2.3. Seguridad.

RNFSeg1. A la hora de guardar las contraseñas de los usuarios en Firestore, estas deben encriptarse.

4.2.2.4. Tiempos de respuesta

RNFTie1. Los tiempos de respuesta de acceso y guardado en base de datos no deben de superar los 15 segundos.

RNFTie2. Los tiempos de búsqueda de la localización, no deben de superar los 2 minutos.

4.3. IDENTIFICACIÓN DE ACTORES DEL SISTEMA

Este sistema cuenta a grandes rasgos con dos tipos de actores que interactuarán con él: usuario registrado y usuario no registrado. A continuación, se describe cada uno de ellos.

4.3.1. Usuario no registrado/ no identificado

Este actor representa a los usuarios que no se han registrado aún en el sistema. La única funcionalidad que podrá utilizar será la de iniciar sesión o registrarse.

4.3.2. Usuario registrado / identificado

Este actor, representa a los usuarios que ya están registrados en el sistema, y que por tanto tienen acceso a toda la funcionalidad de este: Pueden grabar y consultar entrenamientos, ver un histórico de sus datos, hacer un seguimiento de sus pasos y calorías diarias y consultar estimaciones de los tiempos en los que pueden recorrer ciertas distancias.

4.4. ESPECIFICACIÓN DE CASOS DE USO

En este apartado, se detallan los casos de uso de la aplicación, a partir de los requisitos del sistema, especificados en el apartado 4.1.1. *Obtención de los requisitos del sistema.*

Primero, se resume toda la funcionalidad del sistema con un diagrama de contexto. Seguidamente, se especificará cada uno de los casos de uso.

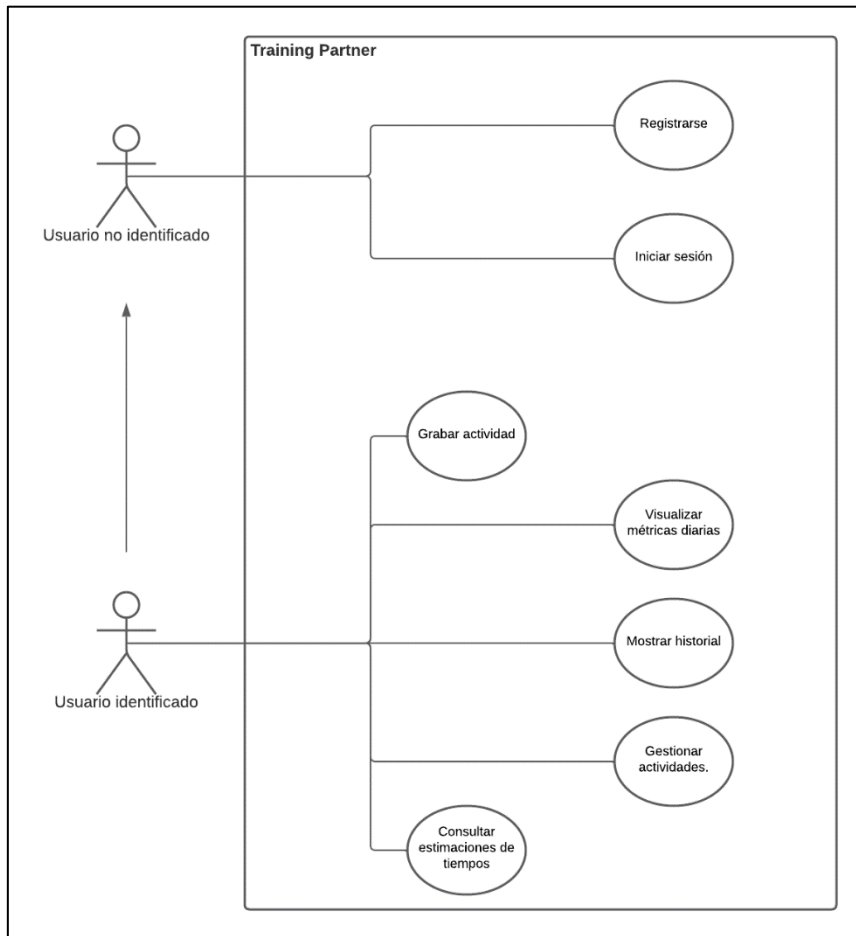


Ilustración 35. Resumen de la funcionalidad del sistema

4.4.1. Casos de uso para usuario no registrado

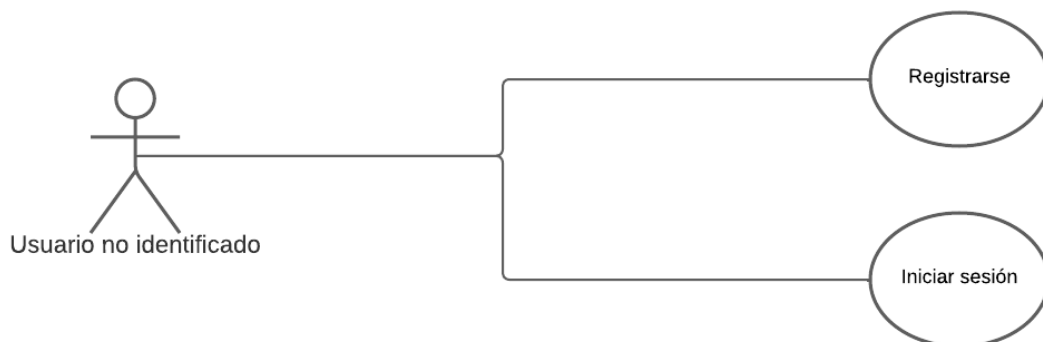


Ilustración 36. Casos de uso para usuario no registrado

Nombre de caso de uso	Requisito
Registro de usuario	RReg1
Descripción	
El usuario no identificado podrá identificarse en la aplicación con un alias que no exista en la base de datos y una contraseña.	

Tabla 40. Caso de uso registro de usuario

Nombre de caso de uso	Requisito
Inicio de sesión	RIni1
Descripción	
El usuario no identificado podrá iniciar sesión en la aplicación siempre que se haya registrado anteriormente, pasando así a ser usuario identificado.	

Tabla 41. Caso de uso inicio de sesión

4.4.2. Casos de uso para usuario identificado

4.4.2.1. Grabar actividades

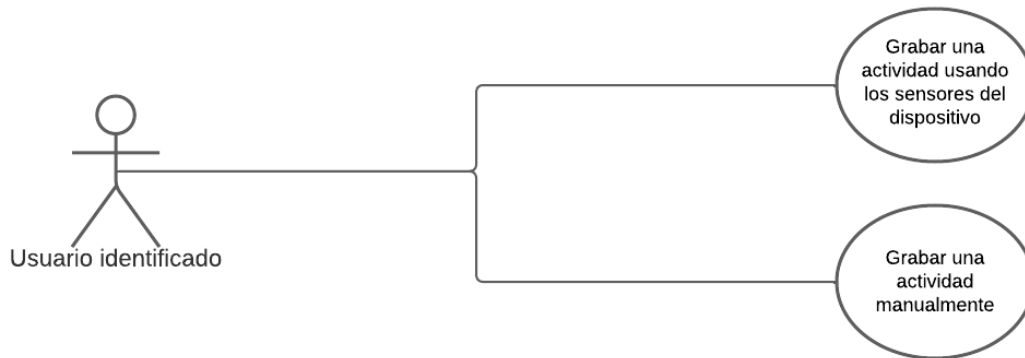


Ilustración 37. Casos de uso grabar actividades

Nombre de caso de uso	Requisito
Grabar una actividad usando los sensores del dispositivo	RGra1
Descripción	
El usuario identificado podrá usar los sensores del dispositivo para poder grabar una actividad de manera automática.	

Tabla 42. Caso de uso grabar actividad con los sensores

Nombre de caso de uso	Requisito
Grabar una actividad de manera manual	RGra2

Descripción
El usuario identificado podrá grabar una actividad introduciendo datos de manera manual.

Tabla 43. Caso de uso grabar actividad manual

4.4.2.2. Visualizar los datos diarios

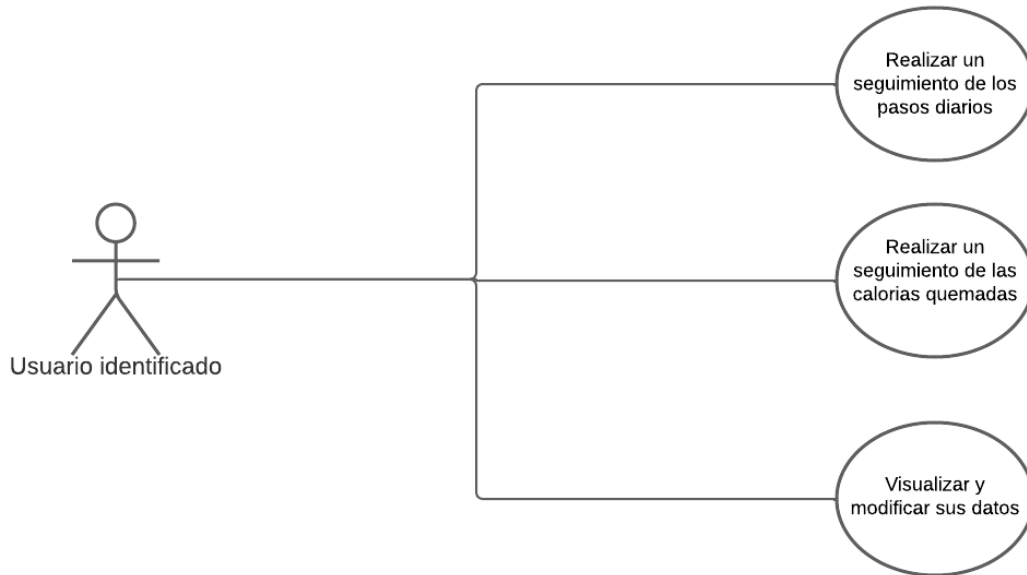


Ilustración 38. Visualizar los datos diarios

Nombre de caso de uso	Requisito
Realizar un seguimiento de los pasos diarios.	RVis1
Descripción	
El sistema contará los pasos que realiza el usuario y este podrá visualizarlos en todo momento.	

Tabla 44. Caso de uso realizar seguimientos pasos

Nombre de caso de uso	Requisito
Realizar un seguimiento de las calorías diarias.	RVis2
Descripción	
El sistema calculará las calorías que quema el usuario y este podrá visualizarlas en todo momento.	

Tabla 45. Caso de uso realizar seguimiento calorías

Nombre de caso de uso	Requisito
Visualizar y modificar sus datos personales.	RVis3
Descripción	
El sistema mostrará al usuario sus datos personales y le permitirá modificar en cualquier momento su altura, su peso y su foto identificativa.	

Tabla 46. Caso de uso visualizar y modificar datos personales

4.4.2.3. Mostrar historial

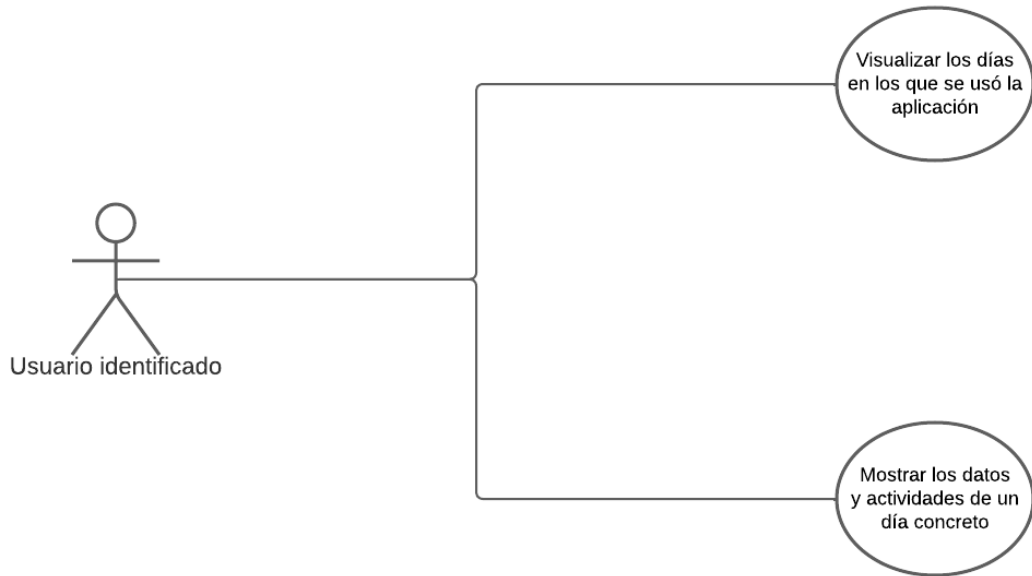


Ilustración 39. Casos de uso mostrar historial

Nombre de caso de uso	Requisito
Visualizar los días en los que se usó la aplicación.	RHis1.1
Descripción	
El usuario podrá ver un calendario en el que se distinguirán los días en los que se usó la aplicación.	

Tabla 47. Caso de uso visualizar días de uso

Nombre de caso de uso	Requisito
Acceder a los detalles de un día pasado.	RHis1.1.2
Descripción	
El usuario podrá acceder mediante un calendario a los días de los que tiene registro. Podrá visualizar los pasos, calorías y actividades que realizó esos días.	

Tabla 48. Caso de uso acceder a los detalles de un día pasado

4.4.2.4. Gestionar actividades

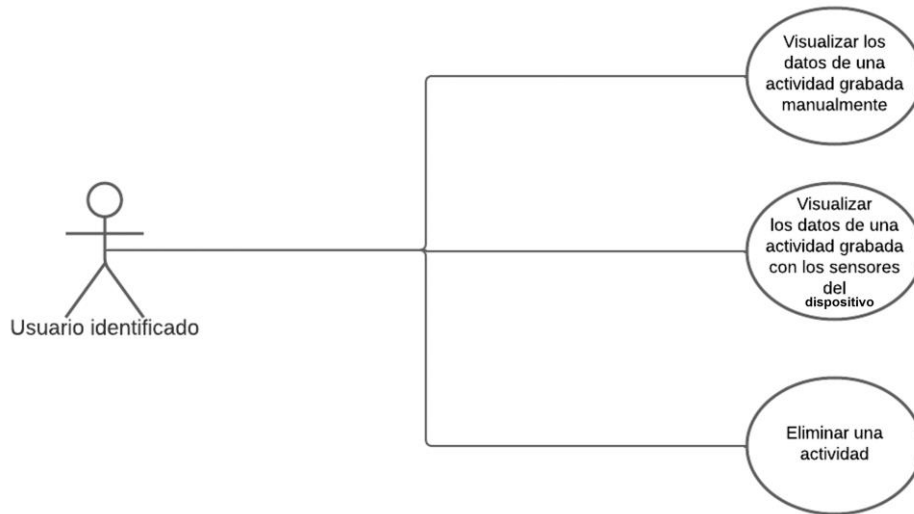


Ilustración 40. Casos de uso gestionar actividades

Nombre de caso de uso	Requisito
Mostrar los detalles de una actividad grabada manualmente.	RGes1.2.
Descripción	
El usuario podrá ver los detalles de una actividad que guardó introduciendo los datos manualmente.	

Tabla 49. Caso de uso mostrar los detalles actividad manual

Nombre de caso de uso	Requisito
Mostrar los detalles de una actividad grabada con los sensores del dispositivo.	RGes1.1.
Descripción	
El usuario podrá ver los detalles de una actividad que guardó haciendo uso de los sensores de su dispositivo.	

Tabla 50. Caso de uso mostrar los detalles actividad automática

Nombre de caso de uso	Requisito
Eliminar una actividad.	RGes2.
Descripción	
El usuario podrá eliminar una actividad finalizada.	

Tabla 51. Caso de uso eliminar actividad

4.4.2.5. Consultar estimaciones de tiempo



Ilustración 41. Casos de uso consultar estimaciones de tiempo

Nombre de caso de uso	Requisito
Consultar las predicciones de tiempo.	RPred1
Descripción	
El usuario podrá visualizar los tiempos en los que el sistema predice que realizará ciertas distancias.	

Tabla 52. Caso de uso consultar predicciones de tiempo

4.5. ASI 3: IDENTIFICACIÓN DE SUBSISTEMAS DE ANÁLISIS

En esta sección descompondremos el sistema en subsistemas para analizarlo de manera más sencilla.

4.5.1. Descripción de los subsistemas

Este prototipo se puede descomponer en 5 subsistemas: subsistema de grabación y visión de actividades, subsistema de métricas y datos diarios, subsistema de predicciones, subsistema de identificación de usuarios y el subsistema de visión del histórico.

4.5.1.1. Subsistema de grabación y visión de actividades

El subsistema de grabación de actividades se encargará de guardar los entrenamientos del usuario ya sea de manera manual o automática. Es competencia de este subsistema todo lo relacionado con la toma, cálculo y visualización de las métricas de las actividades.

4.5.1.2. Subsistema de métricas y datos diarios

Este subsistema será el encargado de calcular y mostrar diariamente los pasos y calorías del usuario de manera automática. También se encargará de la visualización y gestión de los datos personales del usuario: peso, altura, foto personal, nombre y edad.

4.5.1.3. Subsistema de predicciones

Este subsistema se encargará de predecir a través de los datos de los entrenamientos guardados en el sistema, los tiempos que realizará el usuario en ciertas distancias específicas. En el caso que no haya suficientes datos para realizar los cálculos, será competencia de este subsistema notificárselo al usuario.

4.5.1.4. Subsistema de identificación

El subsistema de identificación será el encargado del registro e inicio de sesión en el sistema por parte de los usuarios. Comprobará que todos los datos sean coherentes con los de la base de datos a la hora del registro e inicio de sesión y guardará las credenciales en el sistema.

4.5.1.5. Subsistema de visión del histórico

Este subsistema será el encargado de mostrar el histórico de los días en los que se usó la aplicación. Lo hará a través de un calendario interactivo, que accederá a la base de datos y mostrará para cada día los pasos, calorías y actividades que se realizaron.

4.5.2. Descripción de los Interfaces entre Subsistemas

La comunicación entre subsistemas seguirá el siguiente esquema:

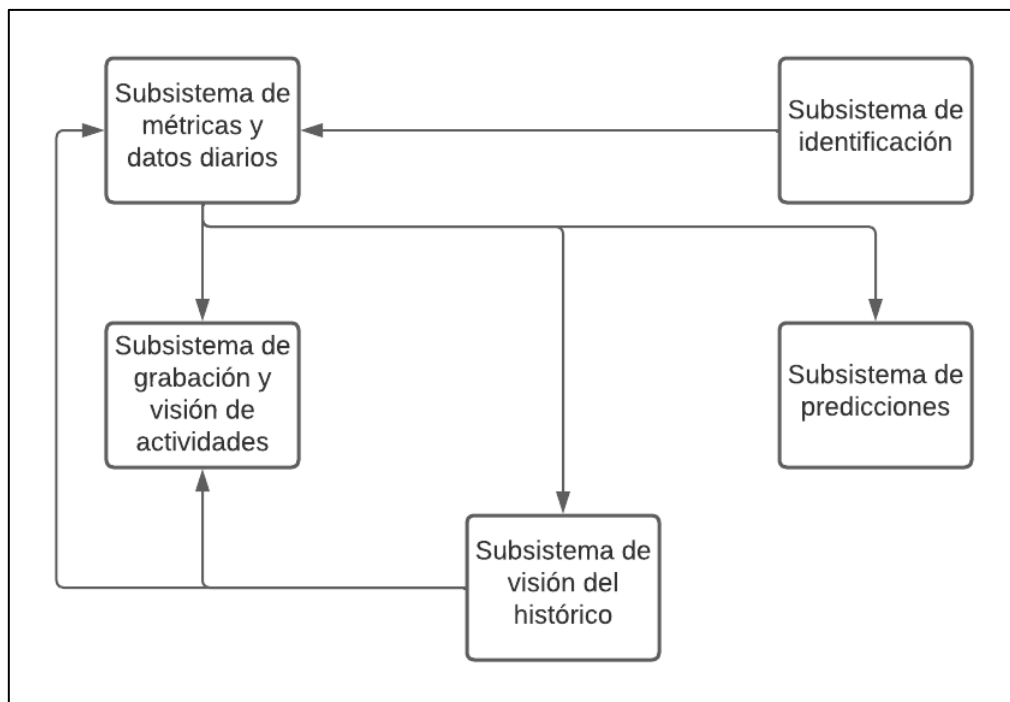


Ilustración 42. Comunicación entre subsistemas

La comunicación entre subsistemas se realizará directamente mediante las interfaces de la aplicación. Además, todos los subsistemas estarán conectados a Firebase, de donde extraerán datos en tiempos real.

4.6. ASI 4: ANÁLISIS DE LOS CASOS DE USO

4.6.1. Caso de uso 1: Registro de usuario

REGISTRO DE USUARIO	
REQUISITO	RReg1
PRECONDICIONES	El usuario no debe de estar registrado en el sistema.
POSCONDICIONES	Se registrarán los datos del usuario en la base de datos y se iniciará su sesión.
ACTORES	Usuario no identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra la ventana de identificación. 2. El usuario introduce los datos de identificación. 3. Si todos los campos están llenos se le permite al usuario pulsar el botón de registro. 4. El sistema comprueba los datos. 5. Si los datos comprobados no son correctos, se muestran los errores y se vacían los campos que tienen fallos. 6. Si los datos son correctos, se guardan en la base de datos, se inicia la sesión del usuario, se guardan sus credenciales y se le dirige a la pantalla principal.
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: La contraseña no cumple con los requisitos de longitud y de caracteres. Se avisa al usuario y no se activa el botón de registro.</p> <p>Escenario alternativo 2: El nombre de usuario ya existe en el sistema. Se volverá al paso 2 del escenario principal.</p>
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 53. Análisis de caso de uso registro de usuario

4.6.2. Caso de uso 2: Inicio de sesión

INICIO DE SESIÓN	
REQUISITO	RIni1
PRECONDICIONES	El usuario debe de estar registrado en el sistema y no debe de tener la sesión iniciada.
POSCONDICIONES	Se guardarán las credenciales del usuario y la sesión quedará iniciada
ACTORES	Usuario no identificado

DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra la ventana de identificación. 2. El usuario accede mediante un botón a la ventana de inicio de sesión. 3. El usuario introduce los datos de inicio de sesión. 4. Si todos los campos están llenos se le permite al usuario pulsar el botón de inicio de sesión. 4. El sistema comprueba los datos. 5. Si los datos comprobados no son correctos, se muestran los errores y se vacían los campos que tienen fallos. 6. Si los datos son correctos, se inicia la sesión del usuario, se guardan sus credenciales y se le dirige a la pantalla principal.
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: La combinación de nombre de usuario y contraseña no existe en la base de datos. Se avisa al usuario y se vuelve al paso 2 del escenario principal.
EXCEPCIONES	Error o falta de conexión: Se muestra un mensaje indicándolo. Error en la base de datos: Se muestra un mensaje indicándolo.

Tabla 54. Análisis de caso de uso de inicio de sesión

4.6.3. Caso de uso 3: Grabar una actividad usando los sensores del dispositivo.

GRABAR UNA ACTIVIDAD USANDO LOS SENSORES DEL DISPOSITIVO	
REQUISITO	RGra1
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	Se guardará la actividad del usuario en la base de datos.
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra la pantalla de grabación automática de actividad. 2. El sistema obtiene la ubicación del dispositivo mediante el sensor GPS. 3. Con la ubicación establecida, el usuario puede iniciar la grabación. 4. El sistema comienza a tomar los datos de la actividad de manera automática y crea una notificación indicándolo. 5. Al finalizar el usuario pulsa el botón de finalización. 5. Los datos de la actividad se guardan en la base de datos. 6. Se redirige al usuario a la pantalla de visualización de la actividad.
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: El usuario pausa la actividad. Se pausa la toma de datos y se vuelve al paso 3 del escenario principal.</p> <p>Escenario alternativo 2: La ubicación tarda demasiado tiempo en establecerse. Se muestra un mensaje del error al usuario y se le redirige a la pantalla principal.</p> <p>Escenario alternativo 3: El usuario no tiene activado el sensor GPS. El sistema pide al usuario que lo active, y lo redirige a los ajustes del dispositivo. Se vuelve al paso 1 del escenario principal.</p> <p>Escenario alternativo 4: El usuario cierra la aplicación mientras se graba la actividad. La aplicación sigue tomando datos en segundo plano.</p>
EXCEPCIONES	Error o falta de conexión: Se muestra un mensaje indicándolo. Error en la base de datos: Se muestra un mensaje indicándolo.

Tabla 55. Análisis de caso de uso de grabar actividad con los sensores

4.6.4. Caso de uso 4: Grabar una actividad de manera manual.

GRABAR UNA ACTIVIDAD DE MANERA MANUAL	
REQUISITO	RGra2
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	Se guardará la actividad del usuario en la base de datos.
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra la pantalla de grabación de actividad manual. 2. El usuario introduce los datos de la actividad realizada. 3. Cuando estén todos los campos llenos, el sistema activa el botón de guardar actividad. 4. El sistema guarda en la base de datos la información de la actividad y calcula el resto de los datos. 5. Se redirige al usuario a la pantalla de visualización de la actividad.
VARIACIONES (ESCENARIOS SECUNDARIOS)	
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 56. Análisis de casos de uso de grabar actividad manual

4.6.5. Caso de uso 5: Realizar un seguimiento de los pasos realizados

REALIZAR UN SEGUIMIENTO DE LOS PASOS REALIZADOS	
REQUISITO	RVis1
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	Los pasos del usuario se irán actualizando y al final del día se guardarán en la base de datos.
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema toma los pasos del usuario mientras este se esté ejecutando en primer o segundo plano en el dispositivo. 2. El usuario podrá comprobar los pasos que lleva a lo largo del día en la pantalla principal del sistema. 3. Al final del día, se guardan los pasos realizados en la base de datos.
VARIACIONES (ESCENARIOS SECUNDARIOS)	
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 57. Análisis de caso de uso de realizar seguimiento pasos

4.6.6. Caso de uso 6: Realizar un seguimiento de las calorías diarias

REALIZAR UN SEGUIMIENTO DE LAS CALORÍAS DIARIAS	
REQUISITO	RVis2
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	Las calorías se irán actualizando y al final del día se guardarán en la base de datos.
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema calcula las calorías del usuario a partir de sus datos, siempre que este se esté ejecutando en primer o segundo plano. 2. El usuario puede comprobar las calorías que lleva quemadas a lo largo del día en la pantalla principal del sistema. 3. Al final del día, se guardan las calorías realizadas en la base de datos.
VARIACIONES (ESCENARIOS SECUNDARIOS)	
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 58. Análisis de caso de uso de realizar seguimiento calorías

4.6.7. Caso de uso 7: Visualizar y modificar los datos personales

VISUALIZAR Y MODIFICAR LOS DATOS PERSONALES	
REQUISITO	RVis3
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	Los datos personales del usuario permanecerán guardados en la base de datos.
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra los datos del usuario en la pantalla principal del sistema. 2. El usuario pulsa en uno de los campos que quiere modificar. 3. El sistema pide al usuario que un nuevo valor para ese campo. 4. El sistema modifica el valor del campo en la base de datos. 5. Los datos se actualizan en la pantalla principal.
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: El usuario modifica su foto. Si no los tiene, el sistema pide al usuario permisos para acceder a los archivos del dispositivo, el usuario puede seleccionar imágenes de la galería o tomarlas directamente con la cámara del dispositivo. Tras seleccionar la imagen, se vuelve al paso 4 del escenario principal.</p> <p>Escenario alternativo 2: EL nuevo valor es vacío o erróneo. Se avisa al usuario y se vuelve al paso 1 del escenario principal.</p>
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 59. Análisis de caso de uso de visualizar los datos personales

4.6.8. Caso de uso 8: Acceder a los detalles de un día pasado

ACCEDER A LOS DETALLES DE UN DÍA PASADO	
REQUISITO	RHis1.1.2.
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra la pantalla de visualización del histórico. 2. El usuario selecciona uno de los días del calendario. 3. Se le redirige a la ventana de información de un día histórico que contiene las actividades grabadas y los datos tomados ese día. 4. Cuando haya terminado de visualizar la información, el usuario pulsa un botón y se le redirige a la ventana de visualización del histórico.
VARIACIONES (ESCENARIOS SECUNDARIOS)	<p>Escenario alternativo 1: El usuario revisa los detalles de una actividad del día seleccionado. Se le redirige a la ventana de visualización de la actividad. Cuando termine de revisarla, se vuelve al paso 3 del escenario principal.</p> <p>Escenario alternativo 2: Se intenta seleccionar un día del que no se tienen datos o que aún no ha ocurrido. El sistema muestra el error y se vuelve al paso 1 del escenario principal.</p>
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 60. Análisis de caso de uso de acceder a los detalles de un día pasado

4.6.9. Caso de uso 9: Mostrar los detalles de una actividad grabada manualmente.

MOSTRAR LOS DETALLES DE UNA ACTIVIDAD GRABADA MANUALMENTE	
REQUISITO	RGes1.2.
PRECONDICIONES	El usuario debe de estar identificado Debe de tener una o más actividades grabadas manualmente
POSCONDICIONES	
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El usuario selecciona una actividad grabada manualmente. 2. El sistema le redirige a la ventana de visualización de actividad grabada manualmente. 3. Se le muestran al usuario los datos de la actividad. 4. Cuando termine de visualizar la actividad, el usuario pulsa un botón que le redirige a la ventana principal del sistema.
VARIACIONES (ESCENARIOS SECUNDARIOS)	
EXCEPCIONES	<p>Error o falta de conexión: Se muestra un mensaje indicándolo.</p> <p>Error en la base de datos: Se muestra un mensaje indicándolo.</p>

Tabla 61. Análisis de caso de uso de mostrar los detalles de una actividad manual

4.6.10. Caso de uso 10: Mostrar los detalles de una actividad grabada con los sensores del dispositivo.

MOSTRAR LOS DETALLES DE UNA ACTIVIDAD GRABADA CON LOS SENSORES DEL DISPOSITIVO	
REQUISITO	RGes1.1.
PRECONDICIONES	El usuario debe de estar identificado Debe de tener una o más actividades grabadas con los sensores del dispositivo
POSCONDICIONES	
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El usuario selecciona una actividad grabada con los sensores del dispositivo. 2. El sistema le redirige a la ventana de visualización de actividad grabada automáticamente. 3. Se le muestran al usuario los datos de la actividad. 4. Cuando termine de visualizar la actividad, el usuario pulsa un botón que le redirige a la ventana principal del sistema.
VARIACIONES (ESCENARIOS SECUNDARIOS)	
EXCEPCIONES	Error o falta de conexión: Se muestra un mensaje indicándolo. Error en la base de datos: Se muestra un mensaje indicándolo.

Tabla 62. Análisis de caso de uso de mostrar los detalles de una actividad automática

4.6.11. Caso de uso 11: Eliminar una actividad

ELIMINAR UNA ACTIVIDAD	
REQUISITO	RGes2
PRECONDICIONES	El usuario debe de estar identificado Debe de tener una o más actividades grabadas.
POSCONDICIONES	La actividad quedará eliminada de la base de datos.
ACTORES	Usuario identificado
DESCRIPCIÓN	<ol style="list-style-type: none"> 1. El sistema muestra la pantalla de visualización de la actividad (es indiferente si ha sido grabada manualmente o con los sensores del dispositivo). 2. El usuario pulsa el botón de eliminar actividad. 3. El sistema pregunta al usuario si está seguro. 4. Si confirma, la actividad se elimina de la base de datos. 5. Se redirige al usuario a la ventana principal del sistema.
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: El usuario no confirma la eliminación de la actividad. No se elimina de la base de datos, y se permanece en la pantalla de visualización del entrenamiento.
EXCEPCIONES	Error o falta de conexión: Se muestra un mensaje indicándolo. Error en la base de datos: Se muestra un mensaje indicándolo.

Tabla 63. Análisis de caso de uso de eliminar una actividad

4.6.12. Caso de uso 12: Consultar las predicciones de tiempo

CONSULTAR LAS PREDICCIONES DE TIEMPOS	
REQUISITO	RPred1
PRECONDICIONES	El usuario debe de estar identificado
POSCONDICIONES	
ACTORES	Usuario identificado
DESCRIPCIÓN	1. El sistema calcula predicciones de tiempos en ciertas distancias a partir de las actividades del usuario. 2. El sistema muestra todas las predicciones en la ventana de predicciones.
VARIACIONES (ESCENARIOS SECUNDARIOS)	Escenario alternativo 1: No hay suficientes actividades guardadas como para calcular predicciones. El sistema en lugar de las predicciones mostrará un mensaje explicativo en la ventana de predicciones.
EXCEPCIONES	Error o falta de conexión: Se muestra un mensaje indicándolo. Error en la base de datos: Se muestra un mensaje indicándolo.

Tabla 64. Análisis de caso de uso de consultar las predicciones de tiempo

4.7. ASI 5: ANÁLISIS DE CLASES

En el siguiente apartado se describen las clases propuestas de cada uno de los subsistemas. Puesto que esto es un análisis previo, es posible que en un futuro se creen o se eliminen algunas de las que se detallan en este apartado.

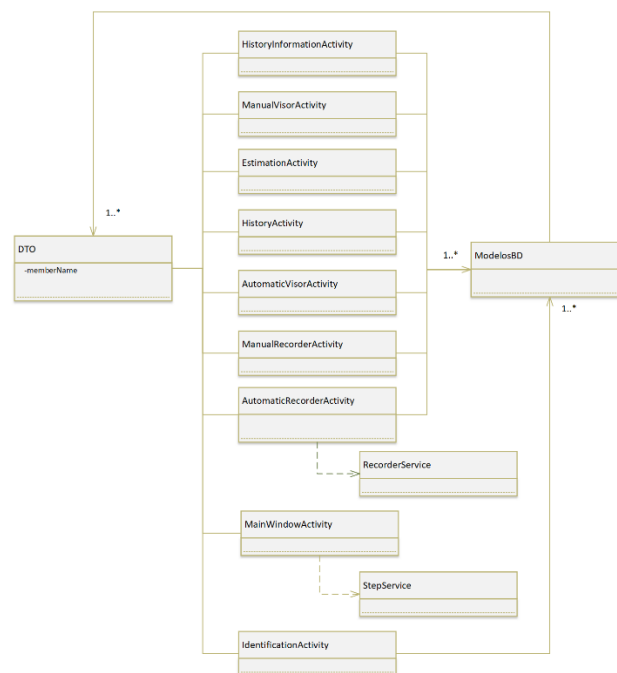


Ilustración 43. Diagrama de clases preliminar

Como se puede comprobar, la aplicación sigue un patrón modelo, vista controlador. Se usarán las pantallas escritas en XML a modo de vista, los Activity a modo de controlador y se crearán modelos para acceder a la base de datos.

DTO y ModelosBD agrupan varias clases que tienen la misma funcionalidad.

4.7.1. Descripción de las clases

4.7.1.1. Clases comunes

Nombre de la clase
DTO
Descripción
Son las clases que sirven para comunicar los modelos de la base de datos con los controladores.

Tabla 65. Descripción DTO

Nombre de la clase
ModelosBD
Descripción
Conjunto de clases que comunican la base de datos con la aplicación. Realizan operaciones de lectura y escritura de datos. Usan los DTO para pasar los resultados a los controladores.

Tabla 66. Descripción ModelosBD

4.7.1.2. Subsistema de métricas y datos diarios

Nombre de la clase
MainWindowActivity
Descripción
Controlador para la pantalla principal, será la encargada de mostrar los datos diarios del deportista y las actividades que ha realizado.
Atributos propuestos
Referencias a los componentes de la vista DeportistaDTO: DTO del usuario actual. ActividadesDeportista: Lista de las actividades que ha realizado ese día el usuario. Pasos: Pasos que lleva el deportista. Calorías: Calorías quemadas por el deportista.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. CambiarPeso, CambiarAltura y CambiarFoto: Métodos usados para modificar los datos del usuario. InicializarMedidas: Método que inicializa el servicio en segundo plano de cuenta de pasos. CalcularCalorias: Método que a partir de los datos del usuario, calcula las calorías quemadas. GuardarDatos: Manda a el modelo que guarde todos los datos recopilados ese día.

Tabla 67. Descripción MainWindowActivity

Nombre de la clase
StepService
Descripción
Servicio que contará los pasos que realiza el usuario y que se ejecutará en primer o segundo plano.
Atributos propuestos
Pasos: Cuenta de los pasos que se han realizado hasta el momento. PasosAnteriores: Último dato de los pasos que se guardó en memoria, usado si el servicio se detiene.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. onStartService: Método que se ejecuta cuando la actividad llama al servicio. SumarPasos: Método que a partir del sensor de acelerómetro del dispositivo calcula los pasos que se realizan y los suma al total.

Tabla 68. Descripción StepService

4.7.1.3. Subsistema de identificación

Nombre de la clase
IdentificationActivity
Descripción
Controlador para la pantalla de identificación, la cual será responsable del registro e inicio de sesión de usuarios.
Atributos propuestos
Referencias a los componentes de la vista Datos de registro o inicio de sesión: Todos los datos del registro del usuario, se les asignará el valor con el que introduzca el usuario.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. Registrar: Método que registra al usuario en la base de datos. IniciarSesión: Método que inicia la sesión del usuario en la base de datos.

Tabla 69. Descripción IdentificationActivity

4.7.1.4. Subsistema de predicciones

Nombre de la clase
EstimationActivity
Descripción
Controlador para la pantalla de predicciones, responsable de realizar predicciones de tiempos en ciertas distancias.
Atributos propuestos
Referencias a los componentes de la vista PredictionsDTO: Dto con los datos para calcular las predicciones.
Métodos propuestos

onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla.
PredecirTiempos: Método que hará las predicciones y las mostrará a la interfaz gráfica.
ConseguirDatos: Pedirá al modelo correspondiente datos para realizar las predicciones.

Tabla 70. Descripción EstimationActivity

4.7.1.5. Subsistema de visión del histórico

Nombre de la clase
HistoryActivity
Descripción
Controlador para la pantalla del historial, gestionará el calendario del historial.
Atributos propuestos
Referencias a los componentes de la vista DaysOfTheMonth: Lista con los días del mes de los que se guarda datos. Month: Mes que se muestra actualmente en la vista. Year: Año que se muestra actualmente en la lista.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. ConseguirDias: Obtiene los días en los que se usó la aplicación en un mes y año concreto.

Tabla 71. Descripción HistoryActivity

Nombre de la clase
HistoryInformationActivity
Descripción
Controlador para la pantalla que muestra un día histórico concreto.
Atributos propuestos
Referencias a los componentes de la vista DeportistaDTO: Objeto con los datos del deportista ese día. Pasos: Pasos realizados por el deportista ese día. Calorías: Calorías quemadas por el deportista en esa fecha. Actividades: Lista de actividades que realizó el deportista ese día.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. ConseguirDatos: Le pide al modelo todos los datos necesarios para un día concreto.

Tabla 72. Descripción HistoryInformationActivity

4.7.1.6. Subsistema de grabación y visión de actividades

Nombre de la clase
AutomaticRecorderActivity
Descripción
Controlador para la pantalla de grabación de actividades con los sensores del dispositivo.
Atributos propuestos

Referencias a los componentes de la vista

Velocidad: Velocidad en kilómetros por hora que lleva el usuario.

Distancia: Distancia total que se ha recorrido en la actividad.

RecorderService: Servicio que en segundo plano realiza las mediciones.

Ritmo: Ritmo actual del usuario en min/km.

Coordenadas: Lista de puntos geográficos por los que ha pasado el usuario.

UbicaciónActual: Punto geográfico en el que se encuentra el usuario actualmente.

Métodos propuestos

onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla.

StartService: Comienza el servicio que toma las mediciones.

StopService: Para el servicio momentáneamente.

Finish: Termina la actividad y le pide al modelo que persista los datos.

Tabla 73. Descripción AutomaticRecorderActivity

Nombre de la clase
RecorderService
Descripción
Servicio que en segundo plano tomará las métricas del usuario, para posteriormente mandárselas a AutomaticRecorderActivity para que las pinte.
Atributos propuestos
<p>Velocidad: Velocidad en kilómetros por hora que lleva el usuario.</p> <p>Distancia: Distancia total que se ha recorrido en la actividad.</p> <p>TimeTask: Cronometro que se ejecutará en un hilo diferente.</p> <p>Ritmo: Ritmo actual del usuario en min/km.</p> <p>Coordenadas: Lista de puntos geográficos por los que ha pasado el usuario.</p> <p>UbicaciónActual: Punto geográfico en el que se encuentra el usuario actualmente.</p>
Métodos propuestos
<p>onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla.</p> <p>onStartService: Método que se ejecuta cuando la actividad llama al servicio.</p> <p>onLocationChange: Método que se ejecuta cuando el sensor GPS percibe un cambio de ubicación.</p> <p>CalcularMétricas: Método que calcula los datos de velocidad, distancia y ritmo para almacenarlas y mandárselas al Activity.</p>

Tabla 74. Descripción RecorderService

Nombre de la clase
ManualRecorderActivity
Descripción
Controlador para la pantalla de grabación de actividades manualmente.
Atributos propuestos
<p>Referencias a los componentes de la vista</p> <p>Tiempo: Tiempo que le tomó al usuario completar la actividad.</p> <p>Distancia: Distancia total que se ha recorrido en la actividad.</p> <p>ActividadManualDTO: Dto que contiene los datos y calcula el ritmo y velocidad medios.</p>
Métodos propuestos

onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla.

GuardarActividad: Método que crea el DTO y le pide al modelo que lo guarde en la base de datos.

Tabla 75. Descripción ManualRecorderActivity

Nombre de la clase
AutomaticVisorActivity
Descripción
Controlador para la pantalla de visión de actividades grabadas con los sensores del dispositivo.
Atributos propuestos
Referencias a los componentes de la vista
AutomaticActivityDTO: DTO extraído de la base de datos con toda la información necesaria de la actividad.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. MostrarDatos: Toma los datos del DTO y los muestra en cada uno de los componentes de la vista. DibujarRuta: A partir de los puntos almacenados, dibuja una ruta en el mapa.

Tabla 76. Descripción AutomaticVisorActivity

Nombre de la clase
ManualVisorActivity
Descripción
Controlador para la pantalla de visión de actividades grabadas manualmente.
Atributos propuestos
Referencias a los componentes de la vista
ActividadManualDTO: DTO extraído de la base de datos con toda la información necesaria de la actividad.
Métodos propuestos
onCreate: Método que indica todo lo que se ejecuta al crearse la pantalla. MostrarDatos: Toma los datos del DTO y los muestra en cada uno de los componentes de la vista.

Tabla 77. Descripción ManualVisorActivity

4.8. ASI 8: DEFINICIÓN DE INTERFACES DE USUARIO

En el siguiente apartado se mostrará un boceto de cada una de las pantallas del sistema. Ha de quedar claro que son prototipos y que la apariencia final del sistema puede variar respecto al que se describe a continuación.

4.8.1. Descripción de la interfaz

4.8.1.1. Prototipo de pantalla de registro

Esta será la primera pantalla que le aparezca al usuario no identificado. En ella tendrá que introducir los datos especificados. El botón de registro no se activará hasta que estén todos los datos rellenos.

Cuando se pulse el botón, se harán las comprobaciones pertinentes y en el caso de que los haya se mostrarán los errores.

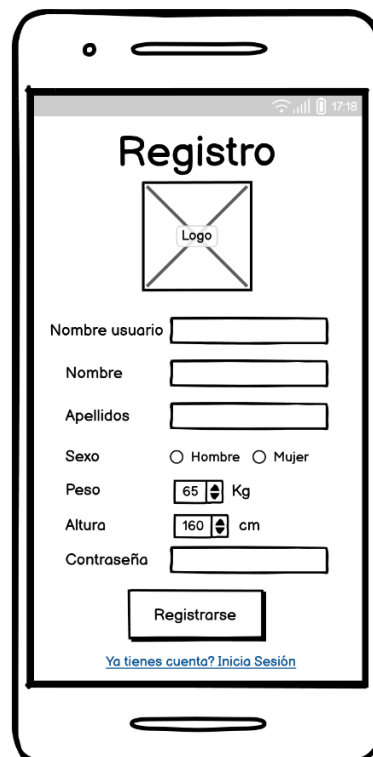


Ilustración de un prototipo de pantalla de registro en un smartphone. La pantalla muestra el título "Registro" y un espacio reservado para un logo. A continuación, se encuentran los campos de entrada para: Nombre usuario, Nombre, Apellidos, Sexo (con opciones Hombre y Mujer), Peso (65 Kg), Altura (160 cm) y Contraseña. Un botón "Registrarse" está situado debajo de los campos. En la parte inferior de la pantalla, hay un enlace que dice "Ya tienes cuenta? Inicia Sesión".

Ilustración 44. Prototipo pantalla de registro

4.8.1.2. Prototipo de pantalla de inicio de sesión

Esta pantalla será la que se le muestre al usuario no identificado cuando indique que ya tiene una cuenta. En ella sólo deberá introducir su nombre de usuario y su contraseña.

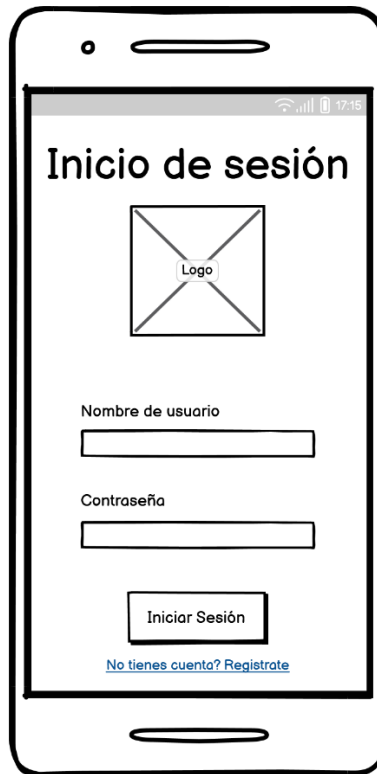


Ilustración 45. Prototipo pantalla de inicio de sesión

4.8.1.3. Prototipo de pantalla principal

La siguiente pantalla será la principal del sistema. En ella se mostrarán todos los datos del usuario y los pasos, calorías y actividades que realiza a lo largo del día.



Ilustración 46. Prototipo pantalla principal

4.8.1.4. Prototipo de pantalla de grabación automática

Esta pantalla se mostrará cuando se comience a grabar una actividad con los sensores del dispositivo. Sólo será accesible cuando el dispositivo establezca la ubicación mediante el sensor GPS, hasta que esto ocurra, se mostrará un mensaje explicativo.



Ilustración 47. Prototipo pantalla de grabación automática

4.8.1.5. Prototipo de pantalla de grabación manual

Pantalla usada para guardar una actividad de manera manual. En ella se detallará la distancia y el tiempo que se realizó.

Para detallar esos tiempos, se usarán sliders a los que se les podrá introducir los datos directamente por teclado para mayor rapidez. El botón de guardar no se activará mientras la distancia o el tiempo sean 0.



Ilustración 48. Prototipo pantalla de grabación manual

4.8.1.6. Prototipo de pantalla de visión de actividades automáticas

Esta es la pantalla que mostrará las actividades grabadas con los sensores del dispositivo. Al tomarse las mediciones de esta manera, se mostrarán también los tiempos de cada uno de los parciales y un mapa interactivo con la ruta seguida.



Ilustración 49. Prototipo pantalla de visión de actividades automáticas

4.8.1.7. Prototipo de pantalla de visión de actividades manuales

Este prototipo de pantalla mostrará las actividades guardadas manualmente. Al no contar con un mapa, mostrará una imagen para hacer más visual la interfaz.

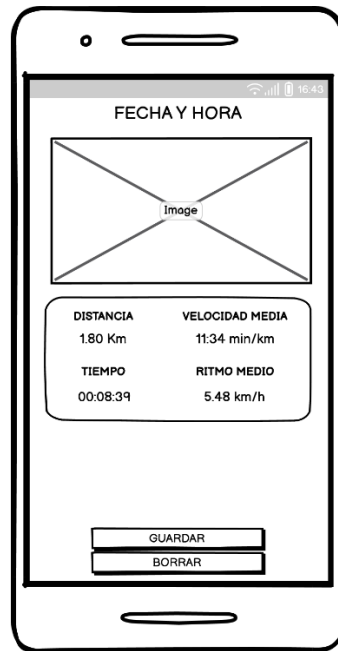


Ilustración 50. Prototipo pantalla de visión de actividades manuales

4.8.1.8. Prototipo de pantalla de visión del histórico

Esta pantalla cuenta con un calendario interactivo en el que el usuario puede seleccionar el día del mes cuya información quiere visualizar.

Podrá además seleccionar el mes y el año que visualizar.

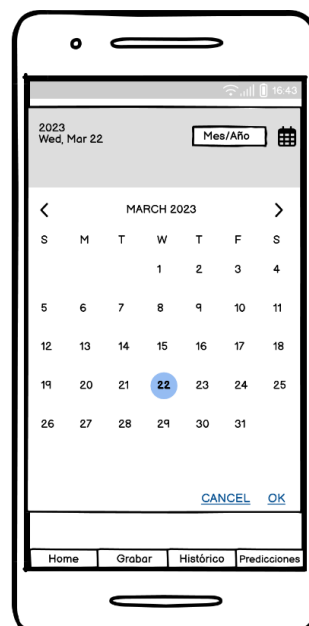


Ilustración 51. Prototipo de pantalla de visión del histórico

4.8.1.9. Prototipo de pantalla de visión de un día histórico

Al seleccionar un día del que existan datos guardados en la pantalla anterior, se mostrará esta pantalla en la que se detallaran los datos que se almacenan de esa fecha.

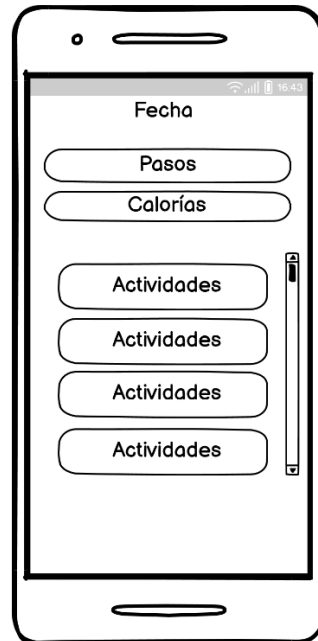


Ilustración 52. Prototipo de visión de un día histórico

4.8.1.10. Prototipo de pantalla de predicciones

Esta pantalla será la encargada de mostrar las predicciones de tiempos del usuario. En el caso de que no haya suficientes datos para calcular esas estimaciones, se le notificará al usuario en esta pantalla.

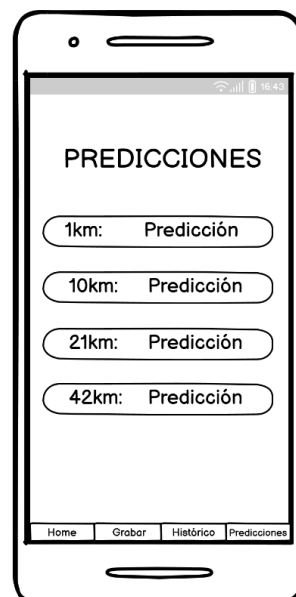


Ilustración 53. Prototipo de pantalla de predicciones

4.8.2. Diagrama de navegabilidad

En el siguiente diagrama se detalla el diagrama de navegabilidad del sistema. Como se puede ver, el sistema está diseñado para permitir al usuario desplazarse entre las principales ventanas del sistema gracias a una botonera inferior. De esta manera, se podrá mover entre pantallas independientemente de cuál sea la de origen.

En el diagrama se puede ver que de las pantallas de grabación (automática y manual), sólo es posible desplazarse a las pantallas de visión de actividades o a la principal (en el caso de que se cancele la grabación de la actividad).

Las ventanas de visión de actividades pueden accederse de tres maneras diferentes:

- A través de la pantalla principal, si hay una actividad grabada ese día
- Tras la grabación de la actividad.
- A través de un día pasado, si se guardaron actividades ese día.

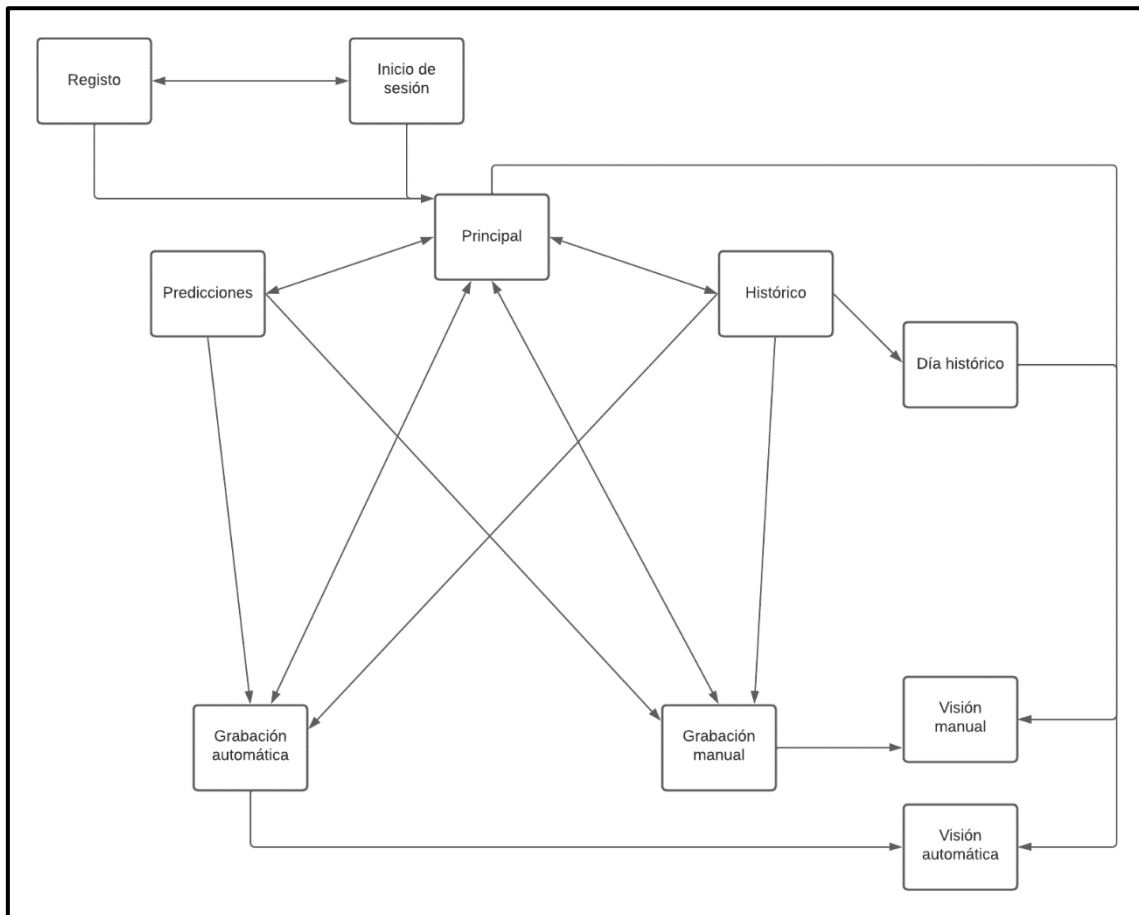


Ilustración 54. Diagrama de navegabilidad

4.9. ESPECIFICACIÓN DEL PLAN DE PRUEBAS

En este apartado se especificarán las pruebas a realizar sobre el prototipo. Con éstas se pretende detectar fallos o defectos en el sistema y asegurar la fácil usabilidad. Se usarán las pruebas



especificadas a continuación para solucionar los errores que se encuentren en la fase de implementación.

4.9.1. Pruebas unitarias

Se usarán este tipo de pruebas para testear el correcto funcionamiento de partes específicas de código. Están enfocadas en encontrar errores en la lógica del sistema y se centran en probar funcionalidades muy concretas. Deben de hacerse basándose en los casos de uso y requisitos descritos en los apartados anteriores.

La naturaleza de este tipo de pruebas permite desarrollarlas y ejecutarlas durante el proceso de desarrollo, sin que el proyecto esté finalizado, permitiéndonos encontrar fallos aislados y fáciles de solucionar.

4.9.2. Pruebas de integración

Se llevarán a cabo tras la realización de las pruebas unitarias. Tienen como objetivo testear la interacción y comunicación de todos los componentes del sistema.

4.9.3. Pruebas de usabilidad

El sistema propuesto en este proyecto está pensado para ser utilizado por diferentes perfiles de usuarios, de distintas edades, sexos, ... Para llegar a ese objetivo, se usan este tipo de pruebas, que consisten hacer que diferentes funcionalidades de la aplicación sean probadas por diferentes perfiles de usuario objetivo.

Los resultados de estas pruebas deben de ser apuntados, y con estos datos se realizará un análisis para encontrar los aspectos que se deben de modificar para asegurar la fácil usabilidad del prototipo.





CAPÍTULO 5: DISEÑO DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

DSI



En este capítulo se va a hablar del diseño y la implementación del sistema.

5.1. DSI 3: DISEÑO DE CASOS DE USO REALES

A continuación, se van a detallar los casos de uso que se consideran más importantes o complejos del proyecto. Cada explicación irá acompañada de su diagrama de secuencia correspondiente.

5.1.1. Caso de uso 3: Grabar una actividad usando los sensores del dispositivo

Se detallará el caso de uso que se menciona en el apartado [4.6.3. Caso de uso 3: Grabar una actividad usando los sensores del dispositivo.](#) usando los sensores del dispositivo.

5.1.1.1. Diagrama de Secuencia

El siguiente es con mucha diferencia el caso de uso más complejo, ya que depende de que el sensor de GPS obtenga correctamente la señal. El usuario accede a la pantalla de grabación de la actividad, si todos los sensores están correctos, podrá comenzar la grabación. Si se perdiera la conexión con el sensor de localización, se le notificaría al usuario. Si esta tardase demasiado tiempo en obtenerse, el sistema redirigiría al usuario a la ventana principal, y la actividad se descartaría.

En el siguiente diagrama se muestra sólo la ejecución del escenario principal. Se usará un servicio en segundo plano para obtener actualizaciones de la ubicación en todo momento (incluso con la pantalla apagada).

El diagrama mostrará una versión simplificada del caso de uso en el que se presupone que no se va a perder la conexión ni la señal GPS en ningún momento, y el usuario solo inicia y finaliza la actividad (sin pausarla). Aunque no se ejemplifiquen el diagrama, todas estas variables están contempladas en el caso de uso.

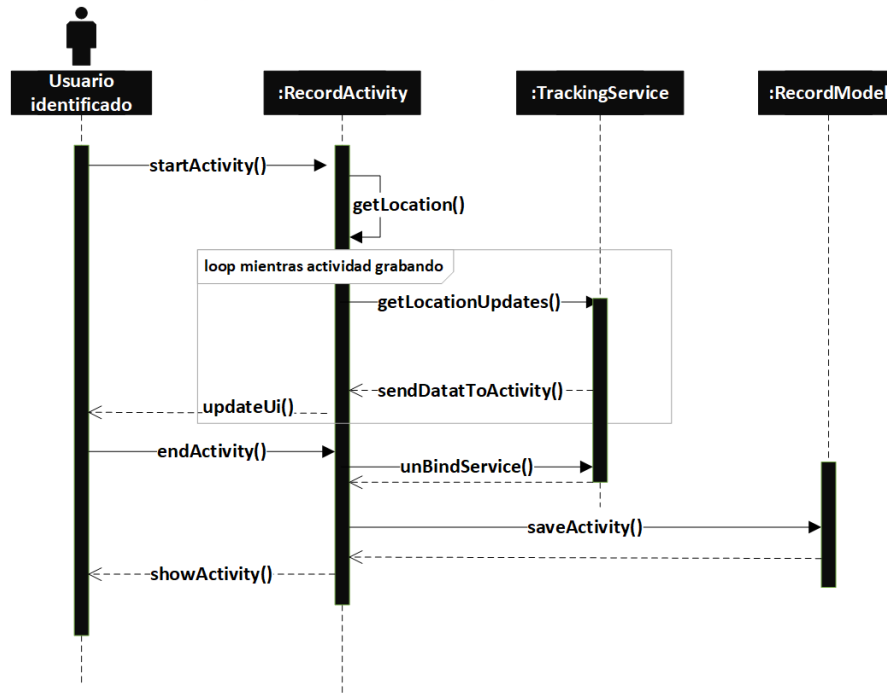


Ilustración 55. Diagrama de secuencia grabar una actividad con los sensores del dispositivo.

5.1.2. Caso de uso 4: Grabar una actividad de manera manual.

A continuación, se detalla el caso de uso indicado en [4.6.4. Grabar una actividad de manera manual](#).

5.1.2.1. Diagrama de secuencia

En el siguiente diagrama se muestra la ejecución del caso de uso de grabar una actividad manualmente. En este caso, el usuario introducirá la distancia que ha realizado y el tiempo que ha tardado y el sistema calculará y guardará el resto de los datos. La interfaz gráfica sólo permitirá guardar la actividad si se han introducido datos en tiempo y distancia.

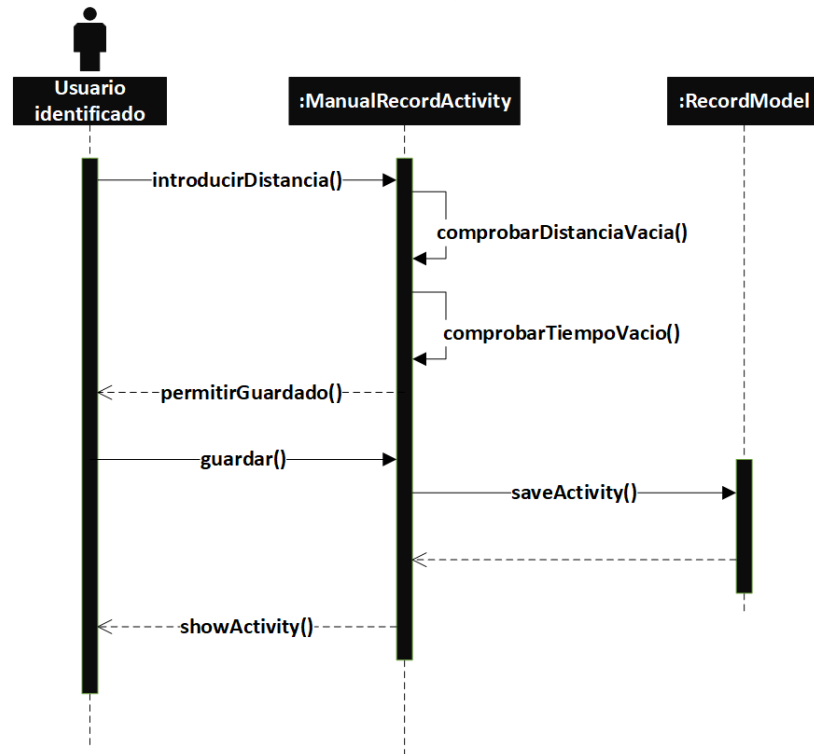


Ilustración 56. Diagrama de secuencia grabar una actividad manual.

5.1.3. Caso de uso 4: Realizar un seguimiento de los pasos realizados.

En este apartado se muestra con mayor detalle el caso de uso indicado en [4.6.5. Realizar un seguimiento de los pasos realizados](#).

5.1.3.1. Diagrama de secuencia.

En el siguiente diagrama de secuencia se detalla la ejecución de este caso de uso. Los pasos del usuario se contarán de manera automática y, en segundo plano, gracias al acelerómetro del dispositivo. Se usarán estos datos de los pasos para calcular las calorías.

Los pasos se guardarán en la base de datos cuando se termine el día o cuando el usuario cierre sesión.

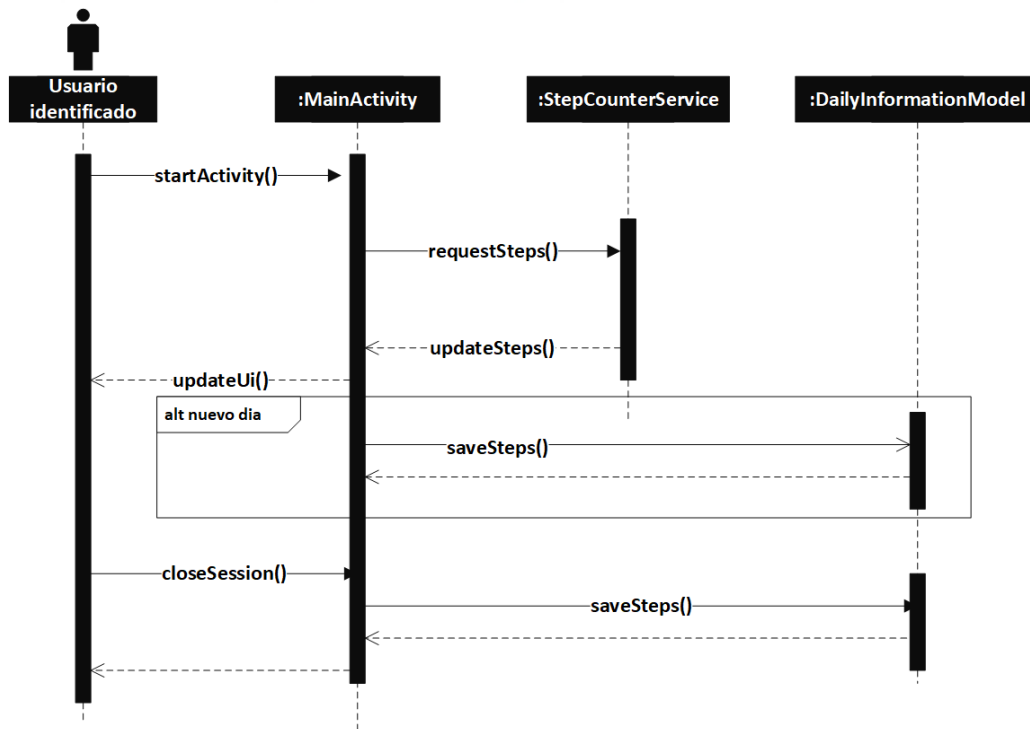


Ilustración 57. Diagrama de secuencia realizar un seguimiento de los pasos realizados.

5.1.4. Caso de uso 9: Mostrar los detalles de una actividad grabada manualmente.

A continuación, se detalla el Caso de uso 9: Mostrar los detalles de una actividad grabada manualmente, detallado en la sección [4.6.9. Caso de uso 9: Mostrar los detalles de una actividad grabada manualmente.](#) A grandes rasgos el funcionamiento general es prácticamente el mismo al visto en el apartado [4.6.10. Caso de uso 10: Mostrar los detalles de una actividad grabada con los sensores del dispositivo.](#)

5.1.4.1. Diagrama de secuencia.

En este caso de uso, el usuario seleccionará una actividad grabada manualmente, el sistema le redirigirá a la ventana de visualización de actividades manuales, donde podrá ver los detalles de su actividad y, si lo deseara, borrarla (aunque esto es competencia del [4.6.11. Caso de uso 11: Eliminar una actividad](#)). Si desea volver a la ventana anterior, podrá hacerlo a través de un botón.

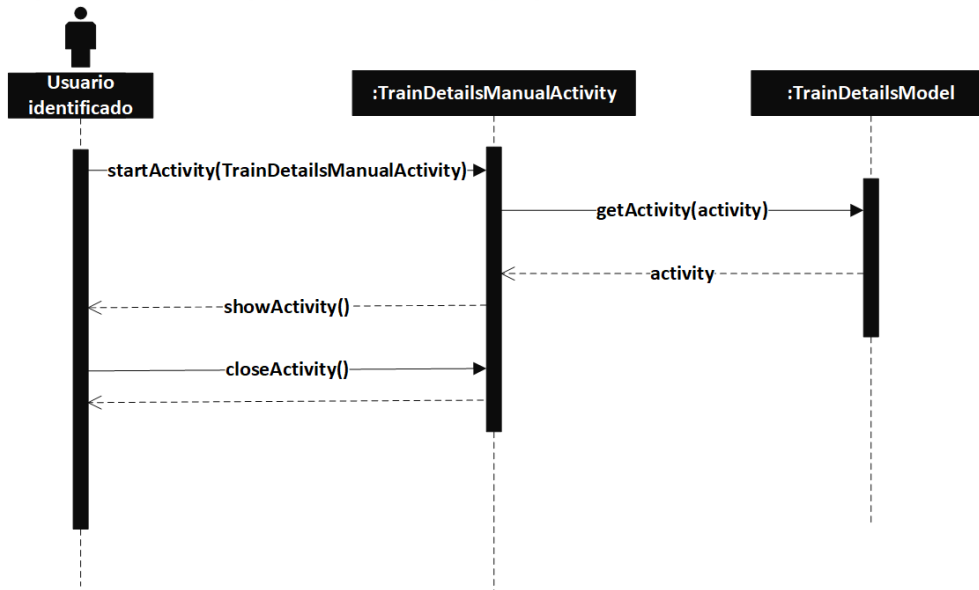


Ilustración 58. Diagrama de secuencia de mostrar una actividad grabada manualmente.

5.1.5. Caso de uso 11: Eliminar una actividad.

En este apartado se muestra el diagrama de secuencia del caso de uso detallado en [4.6.11. Caso de uso 11: Eliminar una actividad.](#)

5.1.5.1. Diagrama de secuencia.

En este caso de uso el usuario se encontrará en la ventana de visualización de la actividad (es indiferente si ha sido grabada manualmente o con los sensores del dispositivo). El usuario pulsará al botón de eliminar actividad y le aparecerá un mensaje preguntándole si está seguro. Si acepta, la actividad se elimina y se le redirige a la ventana principal. Si cancela, se continúa visualizando la actividad.

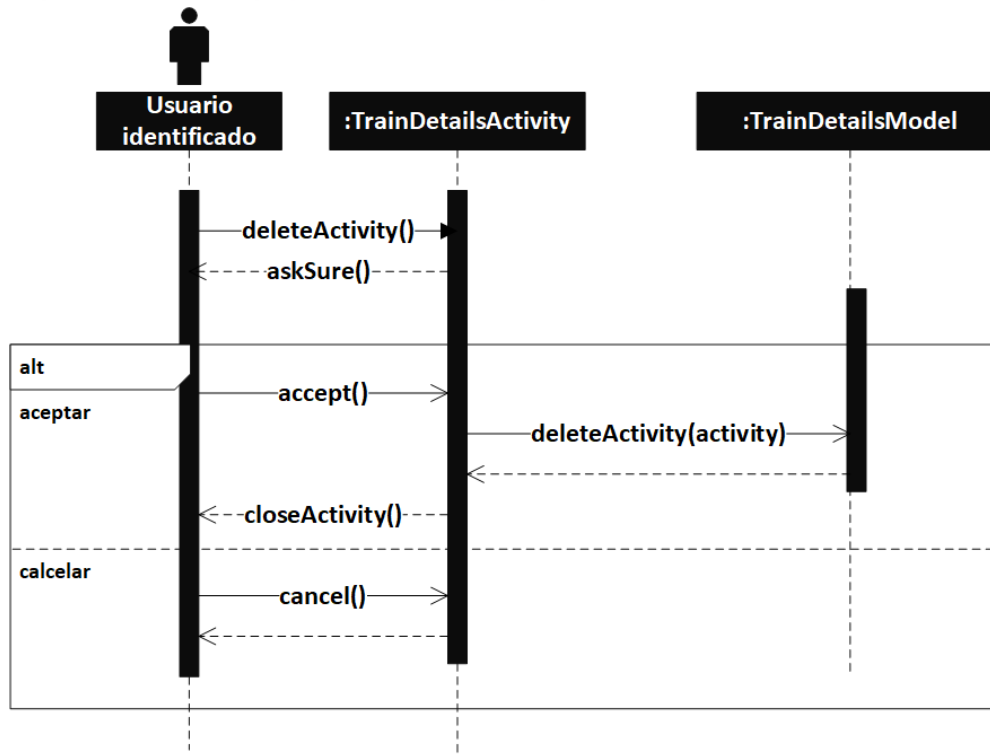


Ilustración 59. Diagrama de secuencia eliminar una actividad.

5.2. DSI 4: DISEÑO DE CLASES

En el siguiente apartado se define y detalla el diagrama de clases.

5.2.1. Diagrama de clases

A continuación, se muestra el diagrama de clases de la aplicación. En primer lugar, se presenta de forma simplificada para clarificar las relaciones entre los paquetes. Seguidamente, se detallan las clases y relaciones de cada uno de los paquetes. Esta estructura persigue facilitar el entendimiento y legibilidad de los diagramas.

En cuanto a los métodos se presentan aquellos con funcionalidad clave, omitiendo métodos tipo getters y setters.

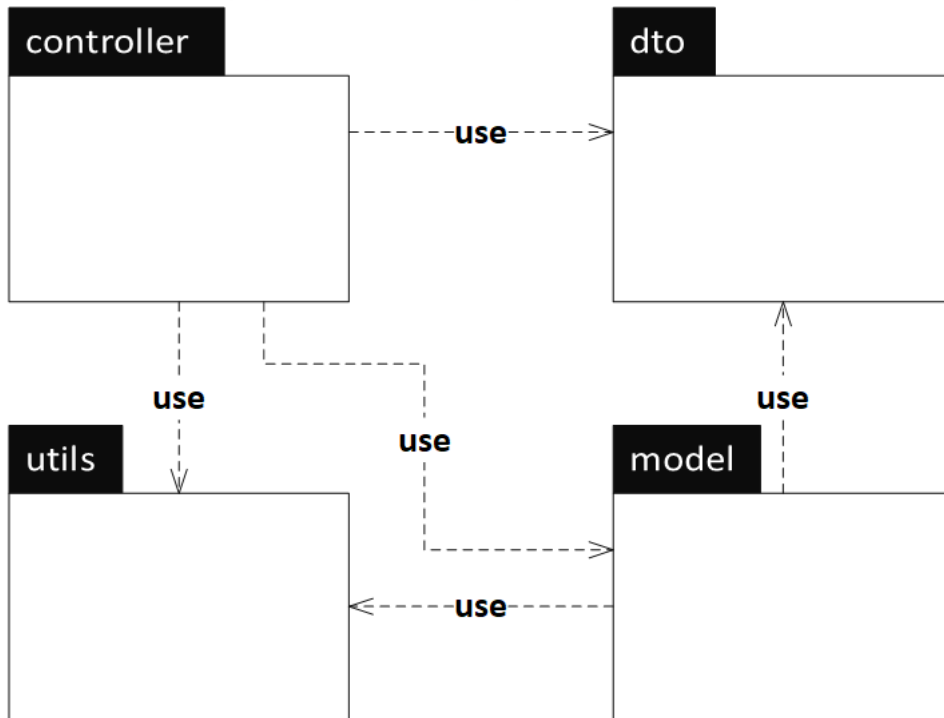


Ilustración 60. Diagrama de clases: comunicación entre paquetes.

Estas son las relaciones entre los distintos paquetes de código Java de la aplicación.

A continuación, se detallan las clases de estos paquetes y las relaciones internas más importantes.

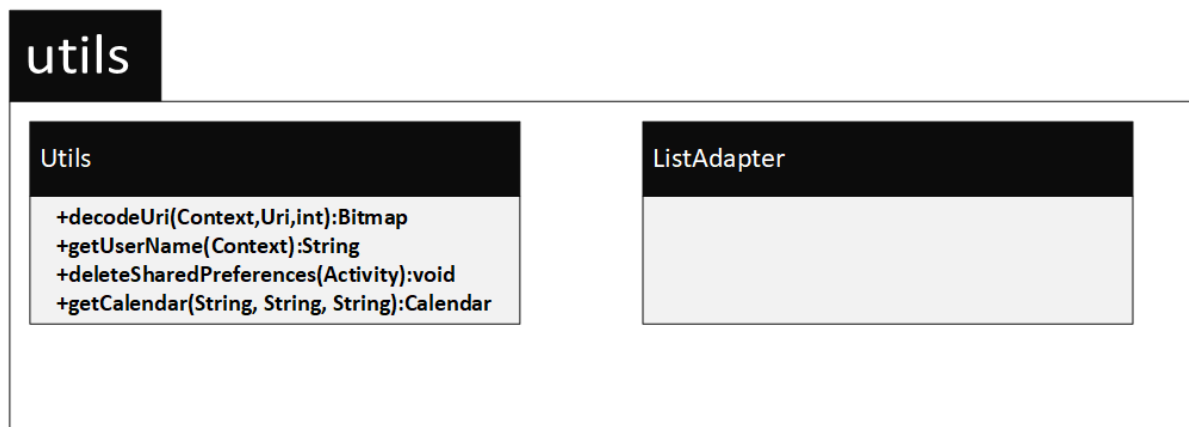


Ilustración 61. Diagrama de clases: Paquete Utils.

dto

activityDto

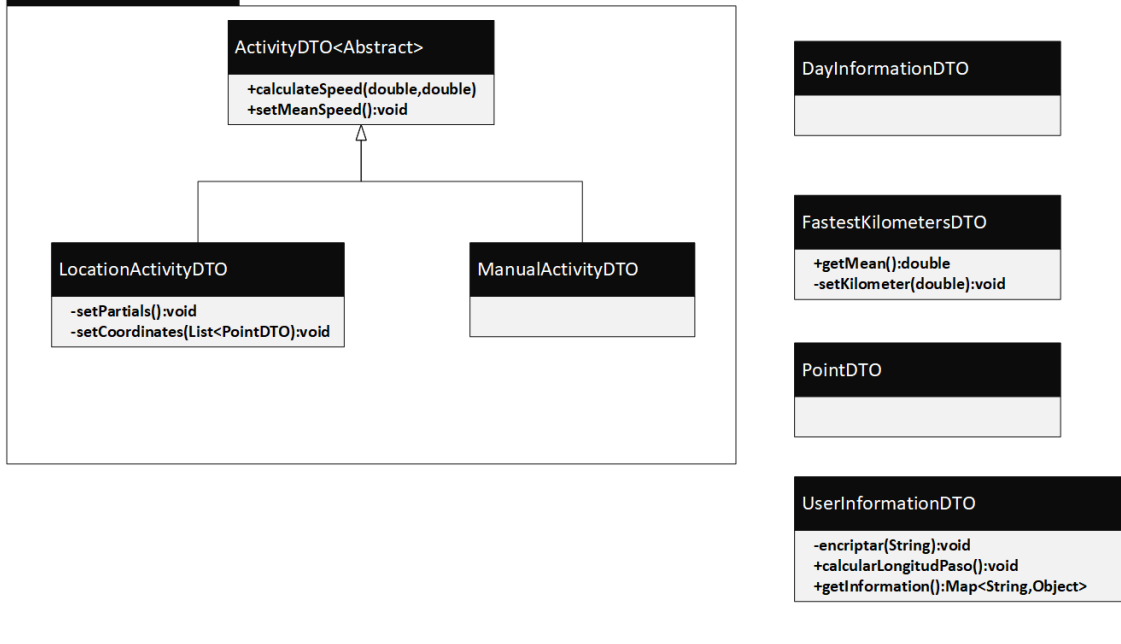


Ilustración 62. Diagrama de clases: Paquete dto.

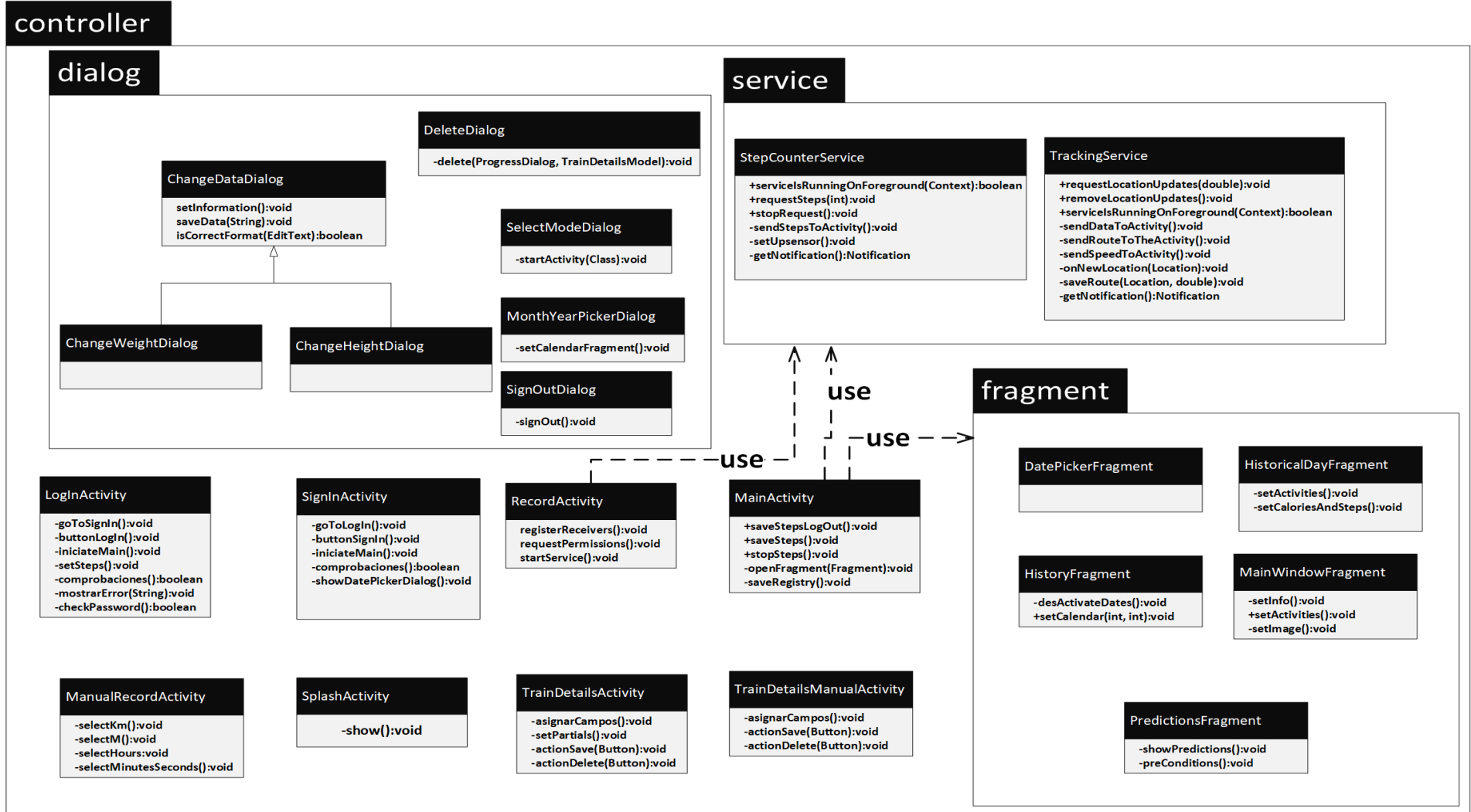


Ilustración 63. Diagrama de clases: Paquete controller.

Model

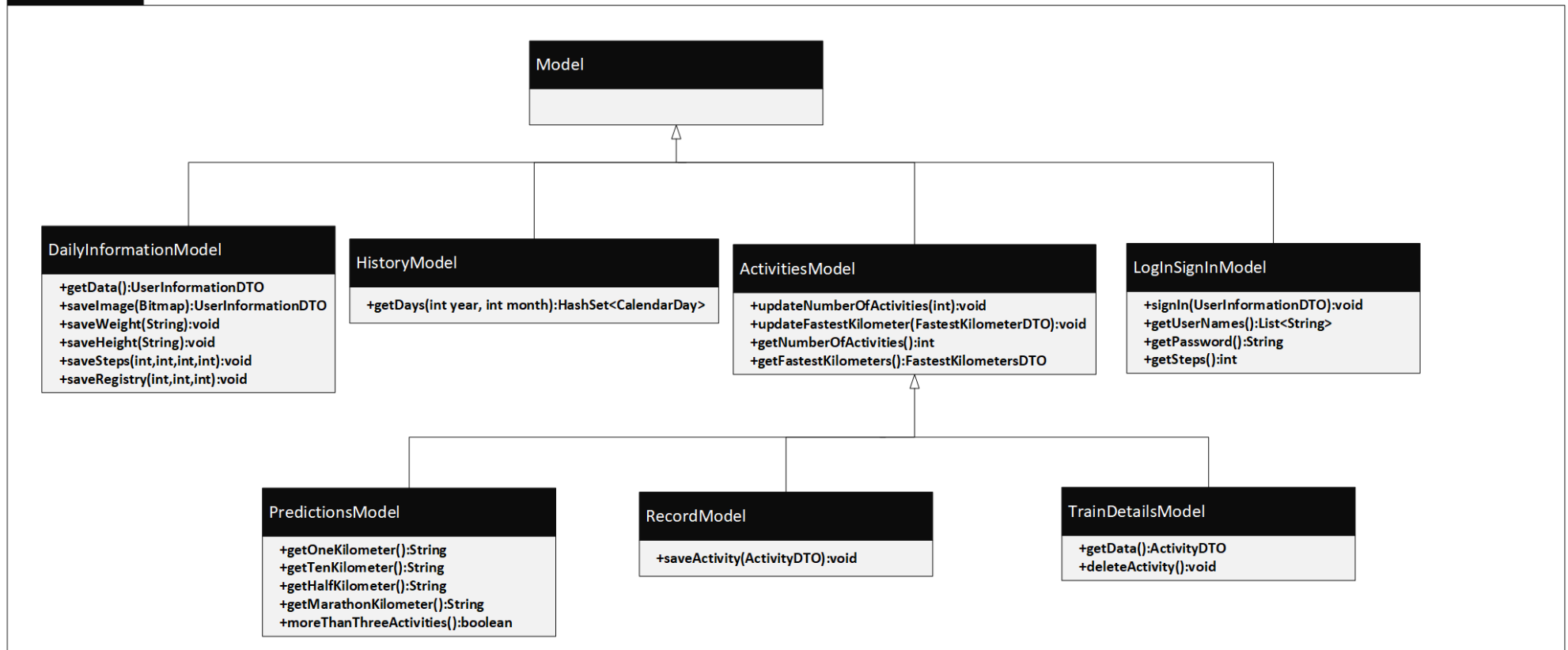


Ilustración 64. Diagrama de clases: Paquete model

5.2.2. Descripción de las clases

Se usará ese apartado para describir las clases que aparecen en los diagramas del apartado anterior. Para cada clase se detallarán los métodos y atributos.

5.2.2.1. Utils

Descripción de las clases del paquete Utils.

Nombre de la clase
Utils
Descripción
Clase usada para brindar funcionalidad común a otras clases del sistema.
Atributos
Métodos
decodeUri: Método encargado de decodificar un URI para convertirlo en Bitmap getUserName: Método que obtiene el userName de SharedPreferences deleteSharedPreferences: Método que elimina toda la información de los ficheros SharedPreferences getCalendar: Método que devuelve un objeto Calendar pasándole el año, mes y día.

Tabla 78. Descripción de la clase Utils.

Nombre de la clase
ListAdapter
Descripción
Clase usada para asignar una lista de actividades a un RecyclerView.
Atributos
list: Lista de ActivityDTO que será asignada al RecyclerView.
Métodos

Tabla 79. Descripción de la clase ListAdapter.

5.2.2.2. Dto

Descripción de las clases del paquete Dto.

Nombre de la clase
ActivityDTO
Descripción
Clase abstracta que representa tanto las actividades manuales como las automáticas.
Atributos
distance: Distancia de la actividad. time: Tiempo total de la actividad. speedMean: Media de velocidad de la actividad. date: Fecha en la que se realizó la actividad. type: Tipo de actividad (manual o automática).
Métodos



calculateSpeed: Método que a partir de la distancia y el tiempo calcula la velocidad.
setMeanSpeed: Método que calcula la velocidad media de la actividad.

Tabla 80. Descripción de la clase ActivityDTO.

Nombre de la clase
LocationActivityDTO
Descripción
Clase que representa las actividades grabadas con los sensores del dispositivo (de manera "automática").
Atributos
Extiende a ActivityDTO, por tanto, tiene todos sus atributos. coordinates: Lista de PointDTO que contiene los puntos por los que pasó el usuario durante la actividad. kilometers: Diccionario con los kilómetros que se recorrieron, y el tiempo que se tardó en realizarlos.
Métodos
setPartials: Ajusta los parciales para cada kilómetro. setCoordinates: Método que asigna los puntos por los que pasó el usuario y los transforma a Point, clase que el mapa puede leer posteriormente.

Tabla 81. Descripción de la clase LocationActivityDTO.

Nombre de la clase
ManualActivityDTO
Descripción
Clase que representa las actividades grabadas manualmente.
Atributos
Extiende a ActivityDTO, por tanto, tiene todos sus atributos.
Métodos

Tabla 82. Descripción de la clase ManualActivityDTO.

Nombre de la clase
DayinformationDTO
Descripción
Clase que representa los valores diarios del usuario (pasos y calorías).
Atributos
steps: Pasos que realiza el usuario en un día. calories: Calorías que quema un usuario en un día.
Métodos

Tabla 83. Descripción de la clase DailyInformationDTO.

Nombre de la clase
FastestKilometersDTO
Descripción

Clase que representa los 3 kilómetros más rápidos realizados por el usuario en actividades de una duración mayor de 3 kilómetros. Estos datos se usan para realizar la predicción de tiempos.

Atributos

kilometer1: Kilómetro más rápido del usuario.

kilometer2: Segundo kilómetro más rápido del usuario.

kilometer3: Tercer kilómetro más rápido del usuario.

changed: Variable booleana que indica si los kilómetros rápidos han cambiado.

Métodos

getMean: Calcula la media de los 3 kilómetros más rápidos.

setKilometer: Asigna un kilómetro entre los 3 más rápidos.

Tabla 84. Descripción de la clase FastestKilometersDTO.

Nombre de la clase
PointDTO
Descripción
Clase que representa un punto geográfico.
Atributos
latitude: Latitud del punto. longitude: Longitud del punto.
Métodos

Tabla 85. Descripción de la clase PointDTO.

Nombre de la clase
UserInformationDTO
Descripción
Clase que contiene toda la información personal del usuario de la aplicación.
Atributos
name: Nombre del usuario. username: Nombre de usuario de la cuenta. birthDate: Fecha de nacimiento del usuario. weight: Peso del usuario. height: Altura del usuario. sex: Sexo del usuario. photo: Ruta a el archivo de foto del usuario. password: Contraseña encriptada del usuario. surname: Apellido del usuario. longitud_paso: Longitud del paso del usuario.
Métodos

encriptar: Método que encripta la contraseña del usuario.
calcularLongitudPaso: Calcula una aproximación del paso del usuario a partir de su altura, edad, peso y sexo.
getInformation: Devuelve un diccionario con la información que se guardará en la base de datos.

Tabla 86. Descripción de la clase UserInformationDTO.

5.2.2.3. Controller

A continuación, se detallarán las clases del paquete controller.

Nombre de la clase
LogInActivity
Descripción
Controlador para el activity de inicio de sesión de usuarios.
Atributos
model: Referencia a un ModelSignInModel. username: Nombre de usuario que se introdujo para iniciar sesión. Password: Contraseña que se introdujo para iniciar sesión. sharedPreferences: Referencia a fichero SharedPreferences.
Métodos
goToSignIn: Método que redirige al usuario a la ventana de registro. buttonLogin: Funcionalidad del botón de inicio de sesión. iniciateMain: Método que inicia la actividad principal, iniciando la sesión y guardando las credenciales del usuario. comprobaciones: Realiza comprobaciones de que los datos son correctos. mostrarError: Muestra un error por pantalla. checkPassword: Comprueba que la contraseña es igual que la guardada en la base de datos (encriptada).

Tabla 87. Descripción de la clase LogInActivity.

Nombre de la clase
SignInActivity
Descripción
Controlador para el activity de registro de usuarios.
Atributos
weight: Peso que ha introducido el usuario al registrarse. height: Altura que ha introducido el usuario al registrarse. name: Nombre que ha introducido el usuario al registrarse. surname: Apellido que ha introducido el usuario al registrarse. username: Nombre de usuario que ha introducido el usuario al registrarse. password: Contraseña que ha introducido el usuario al registrarse. birth: Fecha de nacimiento que introducido el usuario al registrarse. sex: Sexo que ha introducido el usuario al registrarse. model: Referencia a LogInSignInModel. sharedPreferences: Referencia a fichero SharedPreferences.
Métodos



goToLogin: Método que redirige al usuario a la ventana de inicio de sesión.
buttonSignIn: Funcionalidad del botón de registro.
iniciateMain: Método que inicia la actividad principal, iniciando la sesión, guardando las credenciales del usuario.
comprobaciones: Realiza comprobaciones de que los datos son correctos.
showDatePickerDialog: Método que muestra un DatePickerDialog para seleccionar la fecha de nacimiento.

Tabla 88. Descripción de la clase SignInActivity.

Nombre de la clase
SplashActivity
Descripción
Controlador para el activity de la pantalla splash.
Atributos
Métodos
show: Muestra la pantalla splash al inicio de la aplicación.

Tabla 89. Descripción de la clase SplashActivity.

Nombre de la clase
ManualRecordActivity
Descripción
Controlador para el activity de grabación manual de las actividades.
Atributos
pickerKm: Number picker para seleccionar los kilómetros. pickerM: Number picker para seleccionar los metros. pickerHours: NumberPicker para seleccionar las horas que se tardó en realizar la actividad. pickerMinutes: NumberPicker para seleccionar los minutos que se tardó en realizar la actividad. pickerSeconds: NumberPicker para seleccionar los segundos que se tardó en realizar la actividad.
Métodos
selectKm: Método para seleccionar los kilómetros realizados. selectM: Método para seleccionar los metros realizados. selectHours: Método para seleccionar las horas que se tardaron. selectMinutesSeconds: Método para seleccionar los minutos y segundos que se tardaron.

Tabla 90. Descripción de la clase ManualRecordActivity.

Nombre de la clase
RecordActivity
Descripción
Controlador para el activity de grabación automática de las actividades.
Atributos

mService: Referencia a TrackingService.
routeCoordinates: Lista de Point con las coordenadas por las que pasó el usuario.
time: Tiempo que lleva en funcionamiento la actividad.
speed: Velocidad a la que se está moviendo el usuario.
distance: Distancia que lleva recorrida el usuario.
ended: Variable para indicar si la actividad ha terminado.
start, stop, end: Botones para comenzar, parar y terminar la actividad.
model: Referencia a RecordModel.

Métodos

registerReceivers: Registra los BroadcastReceivers que recibirán información de ubicación, distancia, velocidad y tiempo del TrackingService que se está ejecutando en segundo plano.
requestPermissions: Pide permisos al usuario de la ubicación y de ejecución en segundo plano si no los tuviera.
startService: Inicia el servicio en segundo plano, será parado al poner la variable ended a true.

Tabla 91. Descripción de la clase RRecordActivity.

Nombre de la clase
TrainDetailsActivity
Descripción
Controlador para el activity de visión de las actividades automáticas
Atributos
mapView: Objeto MapView que contiene el mapa con la ruta a mostrar. model: Referencia a TrainDetailsModel. dto: LocationActivityDTO de la actividad a visualizar.
Métodos
asignarCampos: Asigna la velocidad media, la distancia, le tiempo y el ritmo medio a los campos de la UI. setPartials: Asigna los parciales a la tabla de parciales de la UI. actionSave: Guarda la actividad y cierra la actividad. actionDelete: Inicia el DeleteDialog para decidir si eliminar o no la actividad.

Tabla 92. Descripción de la clase TrainDetailsActivity.

Nombre de la clase
TrainDetailsManualActivity
Descripción
Controlador para el activity de visión de las actividades manuales.
Atributos
model: Referencia a TrainDetailsModel. dto: ManualActivityDTO que representa la actividad a mostrar.
Métodos



asignarCampos: Asigna la velocidad media, la distancia, le tiempo y el ritmo medio a los campos de la UI.
actionSave: Guarda la actividad y cierra la actividad.
actionDelete: Inicia el DeleteDialog para decidir si eliminar o no la actividad.

Tabla 93. Descripción de la clase TrainDetailsManualActivity.

Nombre de la clase
MainActivity
Descripción
Controlador para la activity principal de la aplicación que contendrá los diferentes fragments.
Atributos
<p>sharedPrefereces: Referencia al fichero SharedPreferences. model: Referencia a DailyInformationModel. dto: UserInformationDTO con la información del usuario activo de la aplicación. bar: ButtonNavigationBar que servirá para cambiar de un fragmento a otro. mService: StepCounterService que contará en segundo plano los pasos del usuario. steps: Pasos del usuario. today: Día actual.</p>
Métodos
<p>saveStepsLogOut: Guarda los pasos al cerrar sesión. saveSteps: Guarda los pasos cuando se inicia la aplicación en un día nuevo. stopSteps: Para el servicio en segundo plano de conteo de pasos, este servicio se inicia automáticamente al crearse la activity. openFragment: Abre un fragmento específico. saveRegistry: Registra en la base de datos el último día que el usuario abrió la aplicación.</p>

Tabla 94. Descripción de la clase MainActivity.

5.2.2.3.1. Fragment

En esta sección se especificarán las clases del paquete controller/fragment.

Nombre de la clase
DatePickerFragment
Descripción
Clase usada para seleccionar una fecha en el registro de usuarios.
Atributos
listener: OnDateSetListener que está escuchando cuando se selecciona una fecha.
Métodos

Tabla 95. Descripción de la clase DatePickerFragment.

Los siguientes fragmentos son los que se muestran sobre la MainActivity.

Nombre de la clase
MainWindowFragment



Descripción
Fragmento que contiene la información personal del usuario y las actividades que ha realizado en el día actual.
Atributos
model: Referencia a DailyInformationModel. dto: UserInformationDTO con la información del usuario activo de la aplicación. activityDTOList: Lista con las actividades que ha realizado el usuario en el día actual.
Métodos
setInfo: Asigna la información del usuario a la interfaz gráfica. setActivities: Asigna las actividades a la interfaz gráfica a través de ListAdapter. setImage: Asigna la imagen del usuario a la interfaz gráfica, descargandola de Storage de firebase.

Tabla 96. Descripción de la clase MainWindowFragment.

Nombre de la clase
HistoryFragment
Descripción
Fragmento que contiene el calendario con los días en los que el usuario usó la aplicación.
Atributos
sharedPreferences: Referencia a el archivo de SharedPreferences. calendarView: Elemento gráfico del calendario. model: Referencia a HistoryModel. selectedDay, selectedMonth, selectedYear: Día, mes y año que se seleccionó en el calendario.
Métodos
desactivateDates: Método que desactiva las fechas de las que no se guarda registro. setCalendar: Método que modifica el mes y el año que se muestra en el calendario a partir de la fecha que se selecciona en el MonthYearPickerDialog.

Tabla 97. Descripción de la clase HistoryFragment.

Nombre de la clase
HistoricalDayFragment
Descripción
Fragmento que muestra la información y las actividades de un día pasado concreto.
Atributos
activityDTOList: Lista de activities que se realizaron ese día. model: Referencia a DailyInformationModel. year, month, day: Día, mes y año del que se está viendo información.
Métodos
setActivities: Asigna a la interfaz gráfica las actividades que se realizaron ese día a través de ListAdapter. setCaloriesAndSteps: Asigna a la interfaz gráfica las calorías quemadas y los pasos dados ese día.

Tabla 98. Descripción de la clase HistoricalDayFragment.

Nombre de la clase

PredictionsFragment
Descripción
Fragmento que muestra las predicciones de los tiempos que puede realizar el usuario en ciertas distancias.
Atributos
model: Referencia a PredictionsModel. one, ten, half, marathon: TextViews en los que se van a mostrar cada una de las predicciones.
Métodos
showPredictions: Método que muestra las predicciones de tiempo del usuario. preConditions: Método que comprueba si se cumplen las precondiciones para realizar las predicciones.

Tabla 99. Descripción de la clase PredictionsFragment.

5.2.2.3.2. Dialog

En esta sección se especificarán las clases del paquete controller/dialog.

Nombre de la clase
ChangeDataDialog
Descripción
Clase abstracta que representa los dialogos para cambiar los datos del usuario.
Atributos
model: Referencia a DailyInformationModel.
Métodos
setInformation: Método que pone el mensaje correspondiente para el dialog. saveData: Método que cambia el dato en la base de datos. isCorrectFormat: Método que comprueba que el formato del dato es correcto.

Tabla 100. Descripción de la clase ChangeDataDialog.

Nombre de la clase
ChangeWeightDialog
Descripción
Dialog para el cambio del peso del usuario.
Atributos
Como extiende a ChangeDataDialog, tiene sus atributos.
Métodos
Extiende todos los métodos de ChangeDataDialog.

Tabla 101. Descripción de la clase ChangeWeightDialog.

Nombre de la clase
ChangeHeightDialog
Descripción
Dialog para el cambio de la altura del usuario.

Atributos
Como extiende a <code>ChangeDataDialog</code> , tiene sus atributos.
Métodos
Extiende todos los métodos de <code>ChangeDataDialog</code> .

Tabla 102. Descripción de la clase `ChangeHeightDialog`.

Nombre de la clase
DeleteDialog
Descripción
Dialog que pregunta al usuario si esta segundo de eliminar la actividad.
Atributos
Métodos
delete: Elimina la actividad, minimiza el dialog y dirige al usuario a la ventana principal.

Tabla 103. Descripción de la clase `DeleteDialog`.

Nombre de la clase
SelectModeDialog
Descripción
Dialog que permite al usuario iniciar una actividad manual o automática.
Atributos
Métodos
startActivity(): Inicia <code>ManualRecordActivity</code> o <code>RecordActivity</code> dependiendo de la decisión del usuario.

Tabla 104. Descripción de la clase `SelectModeDialog`.

Nombre de la clase
MonthYearDialog
Descripción
Dialog que permite al usuario seleccionar el mes y el año del que quiere ver histórico.
Atributos
Métodos
setCalendarFragment: Modifica el calendario de <code>HistoryFragment</code> dependiendo de el año y el mes que seleccione el usuario.

Tabla 105. Descripción de la clase `MonthYearDialog`.

Nombre de la clase
SignOutDialog
Descripción
Dialog que pregunta al usuario si quiere cerrar la sesión.



Atributos
Métodos
signOut: Cierra la sesión, elimina todos los elementos de SharedPreferences y guarda los pasos en la base de datos.

Tabla 106. Descripción de la clase SignOutDialog.

5.2.2.4. Model

Nombre de la clase
Model
Descripción
Clase abstracta de la que heredan todas las clases del modelo.
Atributos
db: Referencia a la base de datos de Firebase. username: Username del usuario activo de la aplicación en ese dispositivo.
Métodos

Tabla 107. Descripción de la clase Model.

Nombre de la clase
DailyinformationModel
Descripción
Clase que gestiona el guardado y obtención de datos de la Firebase referentes a la información diaria del usuario (datos del usuario, foto, pasos, peso y altura, etc.).
Atributos
Al heredar de Model tiene todos sus atributos. dto: Referencia a UserInformationDTO con la información del usuario activo. bitmap: Bitmap con la foto del usuario. storage: Referencia a FirebaseStorage, usada para guardar y obtener la foto del usuario. calendar: Día actual.
Métodos
getData: Método que obtiene toda la información del usuario y la almacena en un UserInformationDTO. saveImage: Método que guarda en FirebaseStorage la imagen del usuario cuando la cambia. saveWeight: Método que guarda el peso del usuario en la base de datos cuando lo cambia. saveHeight: Método que guarda la altura del usuario en la base de datos cuando la cambia. saveSteps: Método que guarda en la base de datos los pasos del usuario. saveRegistry: Método que registra en la base de datos la última vez que el usuario usó la aplicación.

Tabla 108. Descripción de la clase DailyInformationModel.

Nombre de la clase
HistoryModel
Descripción



Clase que gestiona la interacción del historial con la base de datos de Firebase.

Atributos

Al heredar de Model tiene todos sus atributos.

Métodos

getDays: Método que obtiene de la base de datos una lista con los días de los que se guarda registro, para posteriormente poder mostrarlos en el calendario.

Tabla 109. Descripción de la clase HistoryModel.

Nombre de la clase

LogInSignInModel

Descripción

Clase que gestiona las operaciones con la base de datos de Firebase que tienen que ver con el registro e inicio de sesión de los usuarios.

Atributos

Al heredar de Model tiene todos sus atributos

Métodos

signIn: Método que registra un nuevo usuario en la base de datos.

getUserNames: Método que devuelve una lista con el username de todos los usuarios registrados en el sistema.

getPassword: Método que obtiene de la base de datos la contraseña encriptada de un usuario.

getSteps: Método que obtiene los pasos que tiene guardados en la base de datos un usuario, para que si inicia sesión el mismo día que la cerró, comience con ese número de pasos.

Tabla 110. Descripción de la clase LogInSignInModel.

Nombre de la clase

ActivitiesModel

Descripción

Clase que gestiona las operaciones con la base de datos de Firebase que tienen que ver con las actividades del sistema.

Atributos

Al heredar de Model tiene todos sus atributos

Métodos

updateNumberOfActivities: Método que aumenta el número de actividades superiores o iguales a 3 kilómetros que el usuario tiene guardadas en la base de datos.

updateFastestKilometer: Método que aumenta uno de los registros de kilómetro más rápido que el usuario tiene guardado en la base de datos, eliminando el más lento.

getNumberOfActivities: Método que devuelve el número de actividades de longitud mayor o igual a 3 kilómetros que el usuario tiene guardado en la base de datos.

getFastestKilometers: Método que devuelve los 3 tiempos que el usuario tiene guardados como kilómetros más rápidos en la base de datos.

Tabla 111. Descripción de la clase ActivitiesModel.

Nombre de la clase



PredictionsModel
Descripción
Clase que gestiona las operaciones con la base de datos de Firebase que tienen que ver con las predicciones del sistema.
Atributos
Al heredar de Model tiene todos sus atributos.
Métodos
Al heredar de ActivitiesModel tiene todos sus métodos. getOneKilometer: Devuelve la estimación del tiempo en el que el usuario puede realizar un kilómetro. getTenKilometers: Devuelve la estimación del ritmo en el que el usuario puede realizar 10 kilómetros. getHalfKilometer: Devuelve la estimación del ritmo en el que el usuario puede realizar media maratón. getMarathonKilometer: Devuelve la estimación del ritmo en el que el usuario puede realizar una maratón. moreThanThreeActivities: Método que devuelve si el usuario tiene 3 o más actividades de una duración igual o mayor a 3 kilómetros, para realizar las estimaciones.

Tabla 112. Descripción de la clase PredictionsModel.

Nombre de la clase
RecordModel
Descripción
Clase que gestiona las operaciones con la base de datos de Firebase que tienen que ver con la grabación de actividades.
Atributos
Al heredar de Model tiene todos sus atributos.
Métodos
Al heredar de ActivitiesModel tiene todos sus métodos. saveActivity: Guarda la actividad en la base de datos, independientemente de si es una actividad manual o automática.

Tabla 113. Descripción de la clase RecordModel.

Nombre de la clase
TrainDetailsModel
Descripción
Clase que gestiona las operaciones con la base de datos de Firebase que tienen que ver con la obtención y visualización de las actividades.
Atributos
Al heredar de Model tiene todos sus atributos. year,month,day: Año, mes y día de la actividad que se quiere visualizar. actividadDTO: Elemento en el que se almacenarán los datos de la actividad (tanto como si es manual como automática).
Métodos



Al heredar de ActivitiesModel tiene todos sus métodos.

getData: Obtiene de la base de datos todos los datos de la actividad, tanto como si es automática como manual.

deleteActivity: Elimina de la base de datos la actividad.

Tabla 114. Descripción de la clase TrainDetailsModel.

5.3. DSI 5: DISEÑO DE LA ARQUITECTURA DE MÓDULOS DEL SISTEMA

5.3.1. DSI 5.1 Diseño de Módulos del Sistema

5.3.1.1. Diagrama de paquetes

A continuación, se presenta el diagrama de todos los paquetes del sistema. La aplicación está compuesta por dos paquetes principales. El primero *com.example.trainingPartner* es el que contiene toda la lógica de la aplicación, con todas las clases que se han detallado en el apartado anterior, escritas en Java. El otro paquete “res” contiene todos los recursos de la aplicación: imágenes, dialogs, XML, etc.

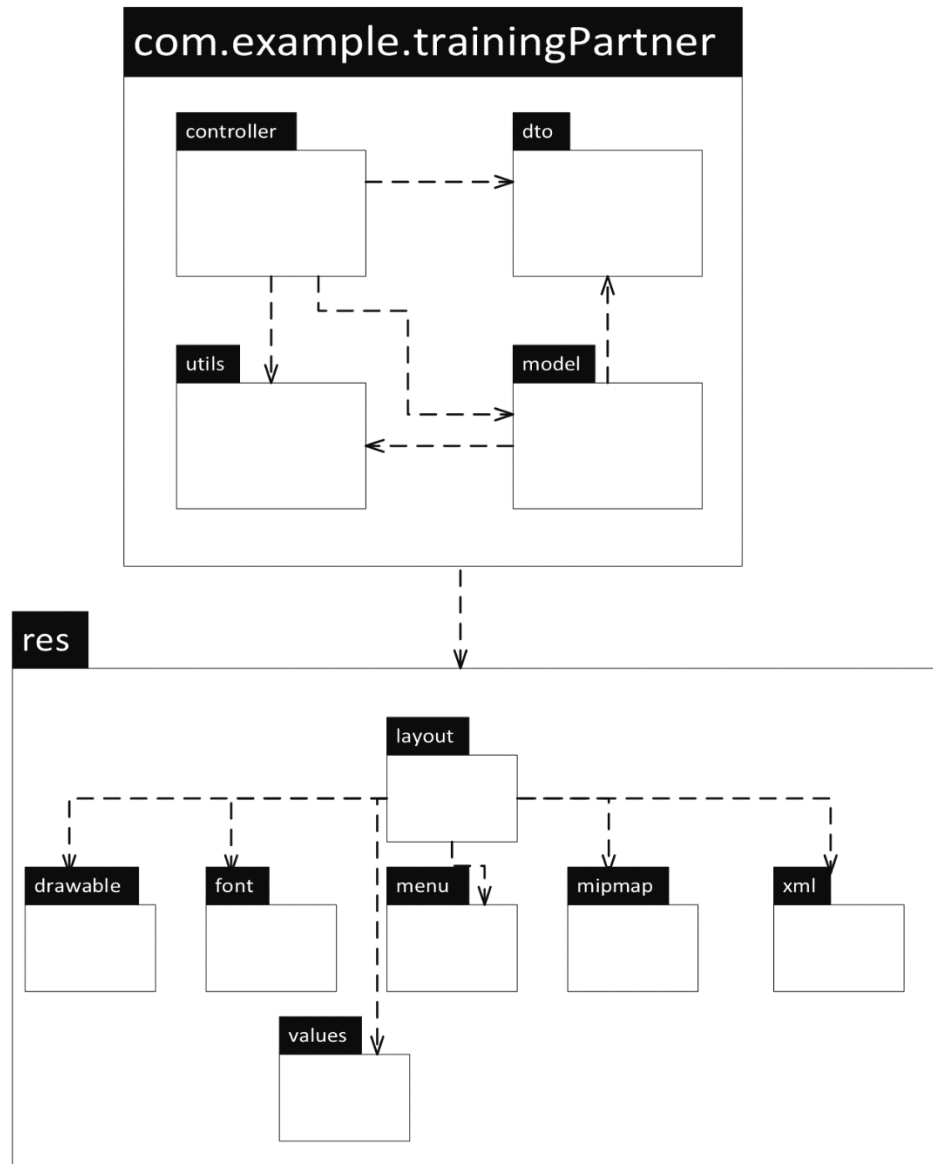


Ilustración 65. Diagrama de paquetes.

A continuación, se detalla el contenido *trainingPartner.controller*.

Este paquete almacena todos los controladores de la aplicación. Las clases contenidas se comunican con las vistas en XML y con los modelos que acceden a la base de datos.

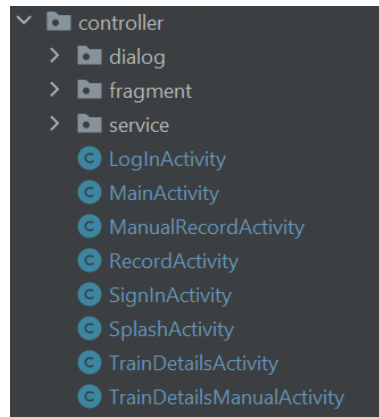


Ilustración 66. Contenido trainingPartner.controller.

Dentro de este paquete hay tres sub-paquetes que se explicarán a continuación.

5.3.1.1.1. Paquete com.example.trainingPartner.controller.dialog

En este sub-paquete se almacenan todos los controladores de los custom-dialogs de la aplicación.

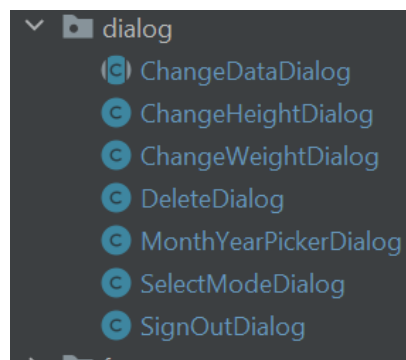


Ilustración 67. Contenido trainingPartner.controller.dialog.

5.3.1.1.2. Paquete com.example.trainingPartner.controller.fragment

Este sub-paquete aloja todos los fragmentos que se van a mostrar en el MainActivity y que ejecutaran gran parte de la funcionalidad de la aplicación.

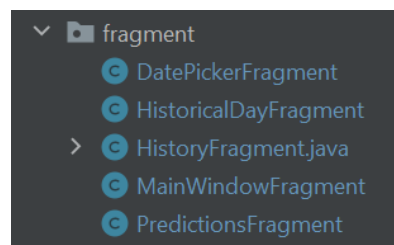


Ilustración 68. Contenido trainingPartner.controller.fragment.

5.3.1.1.3. Paquete com.example.trainingPartner.controller.service

Este sub-paquete contiene los dos servicios que se usan en la aplicación y que deben de ejecutarse en segundo plano. El servicio step-counter que se ejecuta mientras MainActivity no se cierre, y el servicio TrackingService que es el encargado de recoger las métricas de las actividades en segundo plano. Ambos servicios se ejecutan con la aplicación cerrada, o la pantalla del móvil apagada.

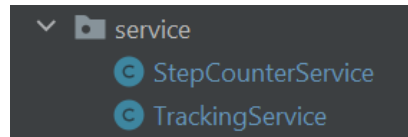


Ilustración 69. Contenido trainingPartner.controller.service.

5.3.1.1.4. Paquete com.example.trainingPartner.model

Este paquete contiene las clases encargadas de interactuar con la base de datos de Firebase. Sólo las clases de este paquete podrán acceder a la base de datos, el paquete controller hará llamadas a este paquete, pero no al contrario.

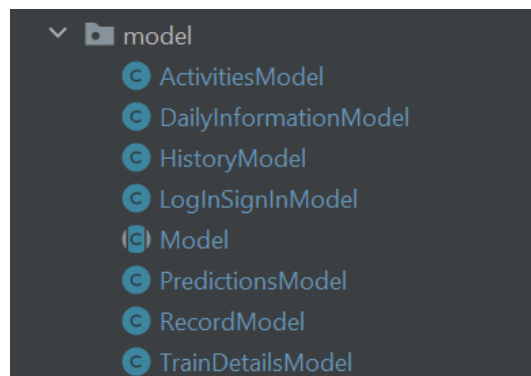


Ilustración 70. Contenido trainingPartner.model.

5.3.1.1.5. Paquete com.example.trainingPartner.dto

Este paquete contiene clases que representan DTOs, es decir Data Transfer Object, que sirven para transferir información entre el paquete controller y el paquete model.

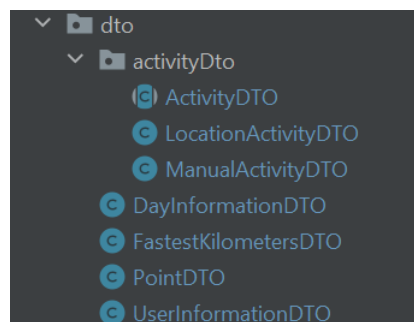


Ilustración 71. Contenido trainingPartner.dto.

5.3.1.1.6. Paquete com.example.trainingPartner.utils

Este paquete aloja las clases que brindan funcionalidad extra que es necesaria en varias clases del sistema.

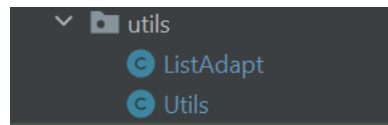


Ilustración 72. Contenido trainingPartner.utils.

5.3.1.1.7. Paquete res.drawable

Este paquete contiene las imágenes usadas en la aplicación, así como los fondos de algunos componentes en XML.

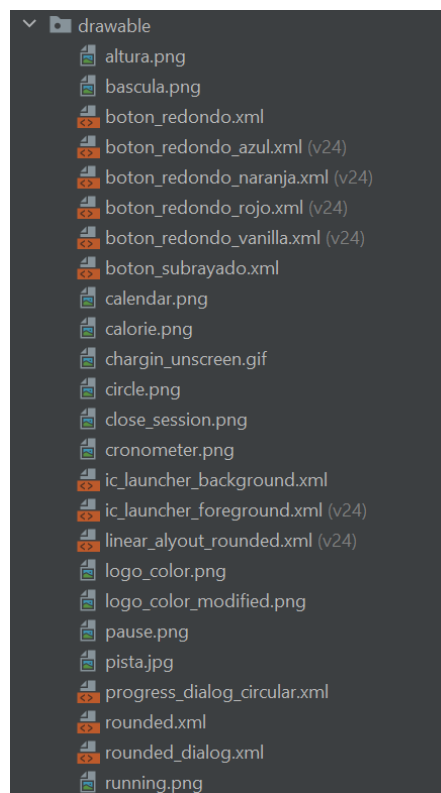


Ilustración 73. Contenido res.drawable.

5.3.1.1.8. Paquete res.font

Este paquete contiene las fuentes de texto.

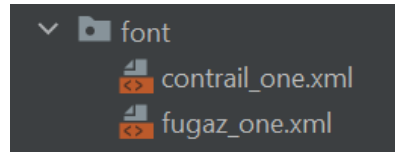


Ilustración 74. Contenido res.font.

5.3.1.1.9. Paquete res.layout

Este paquete contiene todas las pantallas en XML, así como los fragmentos y diálogos.

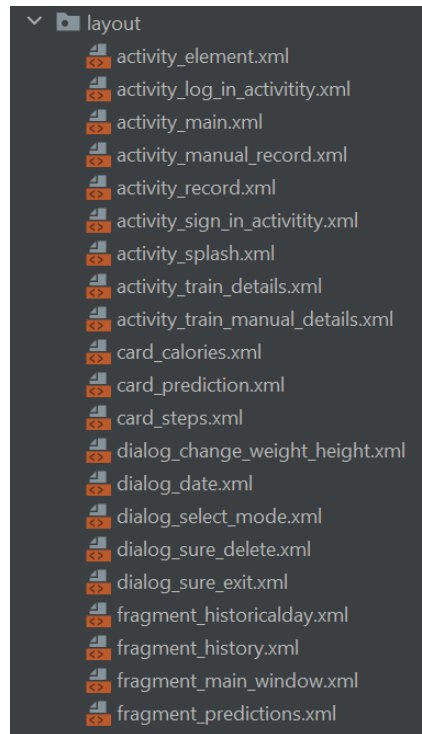


Ilustración 75. Contenido res.layout.

5.3.1.1.10. Paquete res.menu

Este paquete solo contiene una clase, que representa el menú de la barra de selección de fragment que se encuentra en el MainActivity.

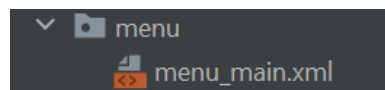


Ilustración 76. Contenido res.menu.

5.3.1.1.11. Paquete res.mipmap

En este paquete están contenidos los diseños del icono de la aplicación.

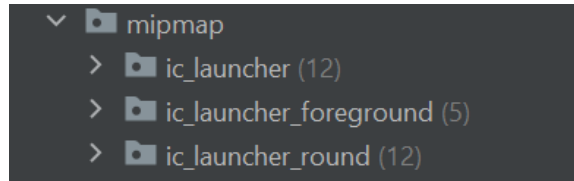


Ilustración 77. Contenido res.mipmap.

5.3.1.1.12. Paquete res.values

Esta carpeta contiene todos los elementos configurables de la aplicación, todos los strings, los colores, temas, etc.

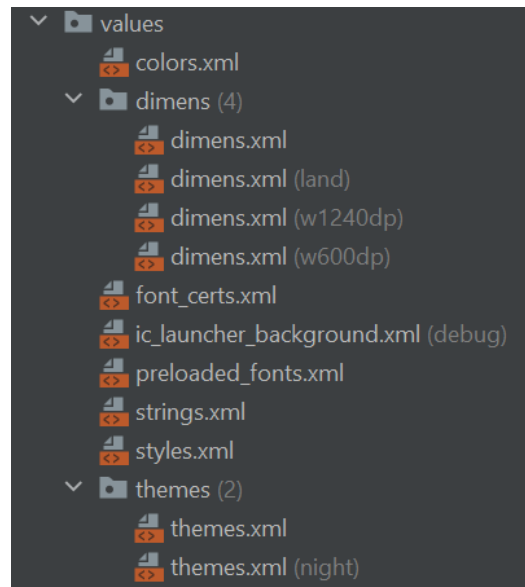


Ilustración 78. Contenido res.values.

5.3.1.1.13. Paquete res.xml

Este paquete almacena el resto de los archivos XML.

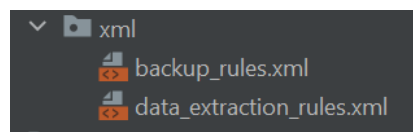


Ilustración 79. Contenido res.xml.

5.3.2. DSI 5.2 Diseño de Comunicaciones entre Módulos

5.3.2.1. Diagrama de componentes

A continuación, se detalla el diagrama de componentes.

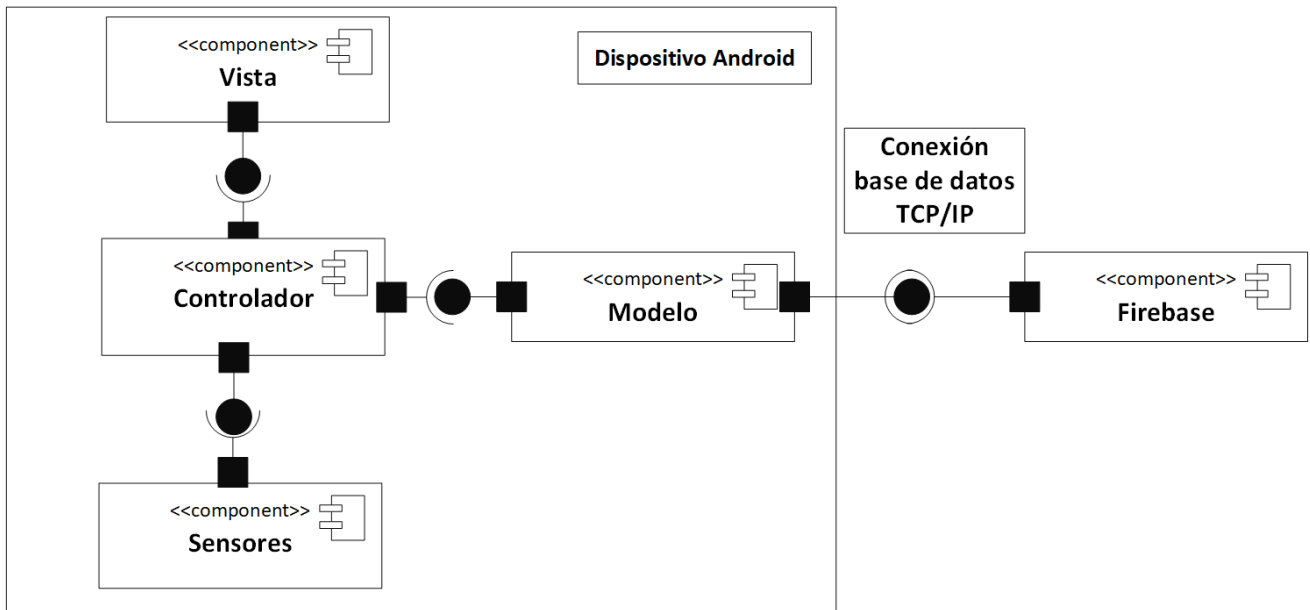


Ilustración 80. Diagrama de componentes.

El diagrama está compuesto por una serie de componentes:

- **Vista:** Todos los elementos que el usuario va a ver.
- **Controlador:** Se refiere a todos los elementos lógicos que controlan el comportamiento de la vista, pero que no se comunican con la base de datos.
- **Sensores:** Los sensores del dispositivo que nos darán información de pasos y localización. (Sensor de acelerómetro y de ubicación).
- **Modelo:** Todos los elementos lógicos cuya única misión es comunicarse con la base de datos.
- **Firebase:** Base de datos en la nube que guarda toda la información del sistema.

5.3.2.2. Diagrama de despliegue

A continuación, se muestra el diagrama de despliegue del sistema.

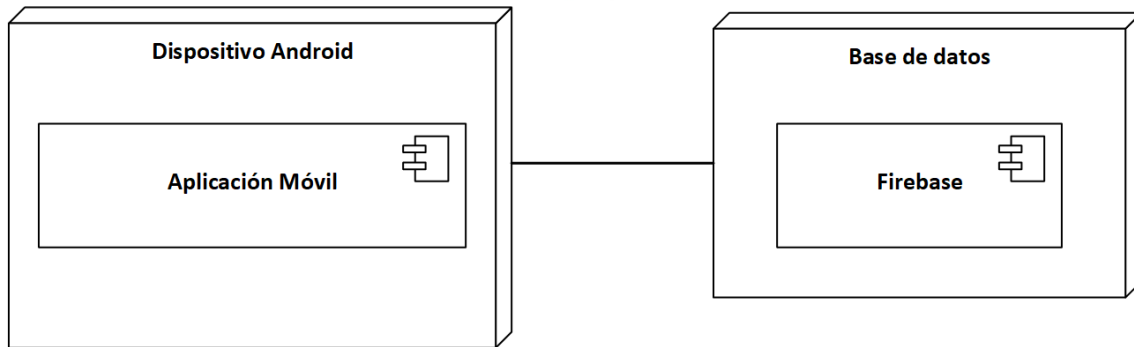


Ilustración 81. Diagrama de despliegue.

En el dispositivo móvil se desplegará la aplicación, mientras que la base de datos será completamente online en Firebase, a través de los servicios de Google.

5.3.3. DSI 5.3 Revisión de la Interfaz de Usuario

En este apartado se va a mostrar el diseño final de las pantallas de la aplicación. La apariencia está basada en la descrita en el apartado [4.8.1. Descripción de la interfaz](#).

Se irán mostrando las pantallas y si es preciso se detallarán aspectos importantes de algunas de ellas.

5.3.3.1. Pantalla de carga.

Esta pantalla será la primera que se le muestre al usuario al iniciar la aplicación. No estaba contemplada en la descripción de la interfaz debido a que fue un añadido posterior. Su principal funcionalidad es la de muestreo del logo de la aplicación mientras se cargan los datos iniciales de la aplicación, para que el usuario no tenga que estar esperando con la pantalla en negro.

El logo de la aplicación fue diseñado mediante un programa online LogoMaker [27].



Ilustración 82. Pantalla de carga.

Este sistema de “Splash Window” se usa en muchas otras aplicaciones como Instagram, Whatsapp o TikTok.

5.3.3.2. Pantalla registro.

Esta pantalla será la segunda que se le muestre al usuario no registrado al abrir la aplicación. Una vez las credenciales del usuario hayan sido guardadas en el sistema, esta ventana se omitirá y se mostrará directamente la pantalla principal.

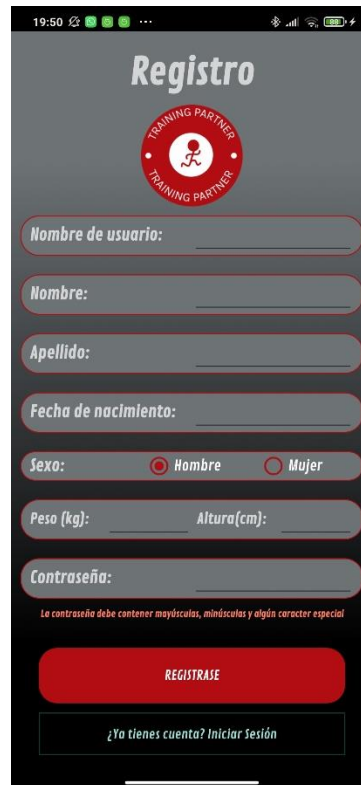


Ilustración 83. Pantalla de registro.

El usuario podrá pulsar el botón de registro en cualquier momento, pero si alguna de las comprobaciones no se satisface, el sistema mostrará el error en la parte inferior de la pantalla.



Ilustración 84. Muestra de error.

Pasados unos segundos el mensaje se ocultará.

Todos los datos pedidos son necesarios para hacer los cálculos de calorías. La altura y el peso se podrán cambiar en cualquier momento.

Pulsando al botón inferior de la pantalla, se redirigirá al usuario a la pantalla de login.

5.3.3.3. *Pantalla de login.*

Esta pantalla servirá para que el usuario con cuenta pueda iniciar su sesión en la aplicación. Sólo se podrá acceder a ella a través de la pantalla de registro, antes de iniciar la sesión hará las comprobaciones pertinentes y mostrará los mensajes de error de la misma manera que la anterior pantalla.

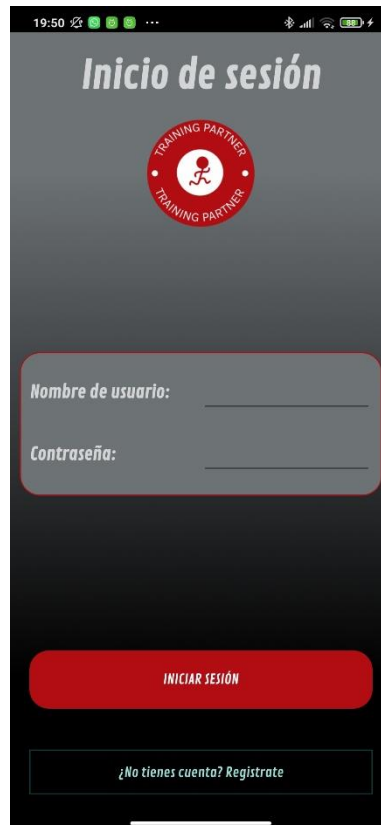


Ilustración 85. Pantalla de LogIn.

El botón inferior redirigirá al usuario a la pantalla de registro.

5.3.3.4. Pantalla principal

Esta pantalla será la que plasme toda la información del usuario y las actividades realizadas ese día. Si hay una sesión abierta en el dispositivo, será la segunda pantalla que se muestre.

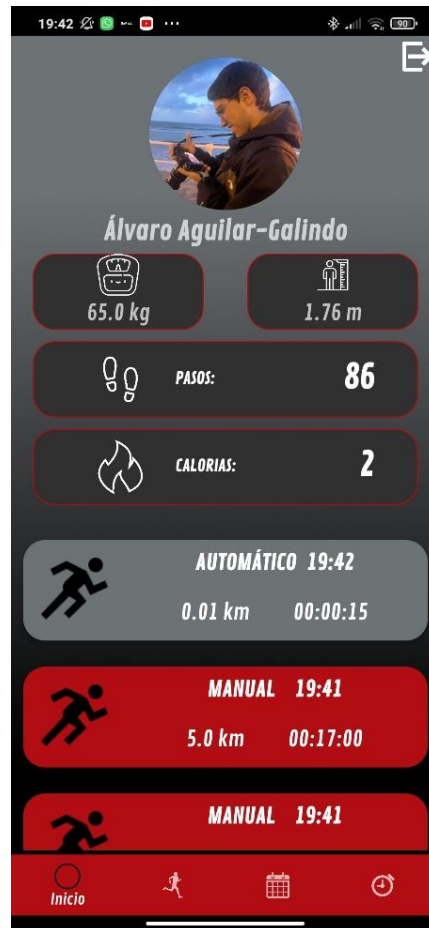


Ilustración 86. Pantalla principal.

En la parte superior de la pantalla se mostrará la imagen del usuario o el siguiente icono si aún no se ha definido una.

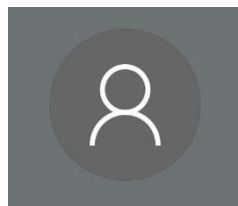


Ilustración 87. Imagen usuario sin foto.

El usuario podrá interactuar en cualquier momento con esta imagen para cambiarla con una guardada en el dispositivo o que se tome en el momento con la cámara.

Bajo la imagen se mostrará el peso y la altura del usuario, ambos paneles de información también son interactivos y permiten cambiar los datos cuando el usuario lo desee.

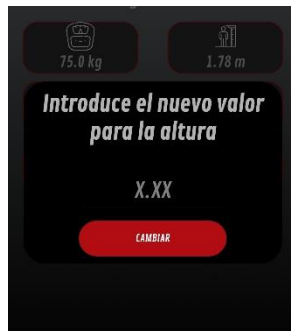


Ilustración 88. Interfaz cambio de altura

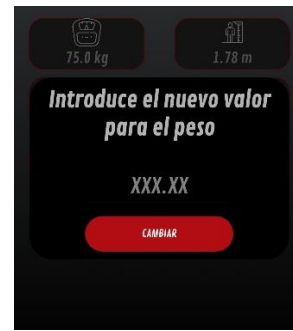


Ilustración 89. Interfaz cambio de peso.

Los campos de pasos y calorías se actualizarán automáticamente gracias al sensor de acelerómetro del dispositivo. Cuando esta pantalla pase del primer plano, o la aplicación se cierre, se mostrará una notificación que irá actualizando los pasos. Esta notificación no se podrá eliminar.

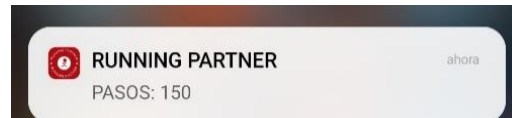


Ilustración 90. Notificación pasos.

Justo debajo de las calorías se colocarán las actividades realizadas ese día. Para diferenciarlas, se mostrarán en gris las grabadas con los sensores del dispositivo y en rojo las manuales.

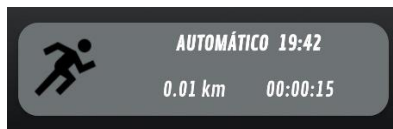


Ilustración 91. Elemento automático.



Ilustración 92. Elemento manual.

Estos elementos mostrarán el tipo de actividad, la hora a la que se realizó, la distancia y el tiempo que se tardó. Son interactivos, y al pulsarlos redirigirán al usuario a los detalles de esas actividades.

El botón de la parte superior izquierda de la pantalla permite al usuario cerrar su sesión y eliminar sus credenciales del sistema. Al pulsarlo se le pedirá confirmación:

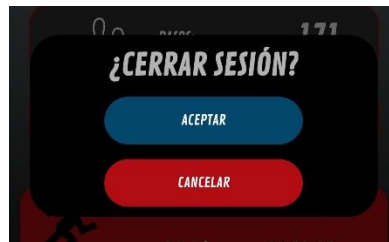


Ilustración 93. Interfaz cerrar sesión.

Finalmente en la parte inferior de la pantalla contiene una barra menú que permitirá al usuario cambiar entre las funcionalidades de la aplicación. Esta manera de interactuar con el sistema es usada en otras aplicaciones deportivas como Strava o Garmin.

5.3.3.5. Pantalla de grabación de actividades con los sensores del dispositivo.

Esta pantalla será la encargada de iniciar, mostrar, pausar y terminar las actividades que el usuario graba usando los sensores del dispositivo.



Ilustración 94. Pantalla de grabación con los sensores.

Se accederá a esta pantalla a través de la barra de menú inferior de la aplicación (presente en la pantalla principal, de historial y de predicciones), pulsando el segundo botón. Se abrirá un dialog que preguntará de qué manera se quiere grabar la actividad, si se selecciona “AUTOMÁTICO”, se abrirá esta pantalla.

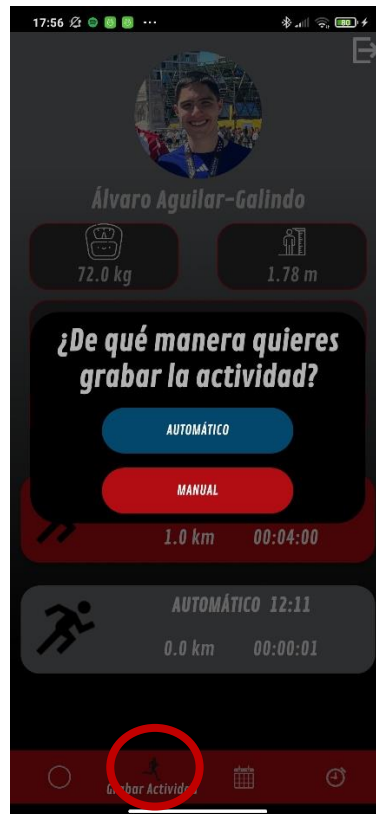


Ilustración 95. Interfaz elección de modo.

En la pantalla se mostrará información continua sobre la distancia, la velocidad actual, el ritmo actual y el tiempo que se lleva grabando la actividad. Si en algún momento se saliera de la aplicación, o se apagara la pantalla del dispositivo, se mostraría una notificación en segundo plano con información actualizada de la localización. Esta notificación no se eliminará hasta que la actividad sea finalizada.

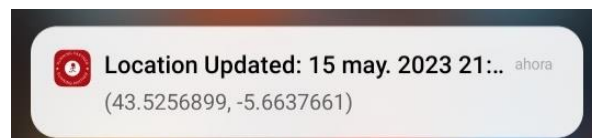


Ilustración 96. Notificación ubicación.

5.3.3.6. Pantalla de grabado manual.

Esta pantalla servirá para introducir manualmente los datos de una actividad y guardarla en el sistema.



Ilustración 97. Pantalla de grabación manual.

El acceso a esta pantalla es igual que a la de grabación con los sensores del dispositivo, a excepción de que se debe de seleccionar “MANUAL” en el dialog emergente.

Para introducir los datos se cuenta con 5 spinners que permitirán seleccionarlos haciendo scroll o introducirlos con el teclado numérico.

Es importante destacar que el botón inferior de GUARDAR permanecerá desactivado hasta que tanto la distancia como el tiempo tengan un valor mayor a 0.

5.3.3.7. Pantalla de visión de una actividad automática.

Al terminar una actividad automática o al interactuar con un elemento de actividad automática, se mostrará esta pantalla que expondrá el recorrido seguido, la distancia, la velocidad media, el tiempo tardado y el ritmo medio de la actividad.



Ilustración 98. Pantalla de visión de actividad automática.

El mapa es interactivo, y mostrará por los que el usuario pasó grabando la actividad. Si se diera el caso de que el usuario hubiese parado y reanudado la actividad moviéndose en el proceso, el mapa también lo mostraría.

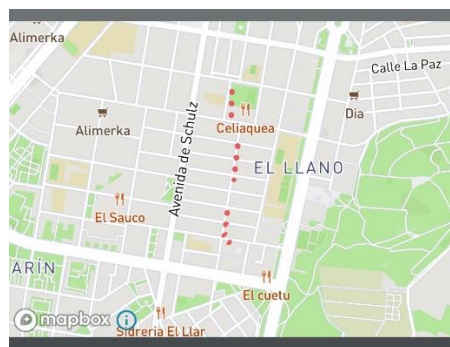


Ilustración 99. Mapa con paradas.

En el caso de que el usuario solo grabara un punto y terminase la actividad en el momento (sin permitir que se realice una línea en el mapa), simplemente se mostrará el lugar en el que se realizó la actividad.

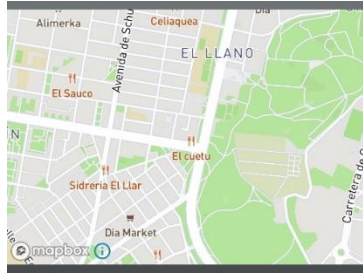


Ilustración 100. Mapa sin puntos.

El botón inferior de Guardar cierra la pantalla y el botón de borrar pregunta al usuario antes de eliminar la actividad.



Ilustración 101. Interfaz borrar actividad.

5.3.3.8. Pantalla de visión de una actividad manual.

El funcionamiento y acceso a esta pantalla es idéntico a la anterior. Mostrará los detalles de las actividades manuales tras grabarlas y al seleccionarlas para ver sus datos.

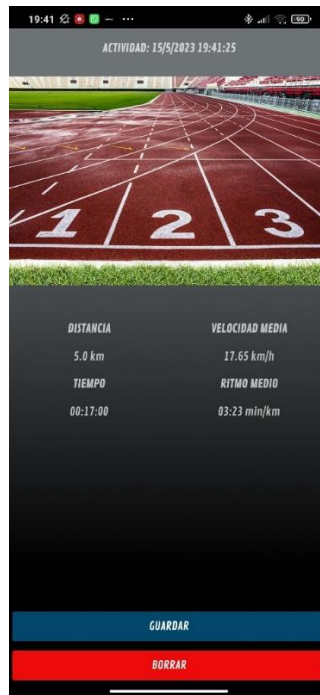


Ilustración 102. Pantalla de visión de actividad manual.

Para cada actividad se muestran las métricas de distancia, velocidad media, tiempo y ritmo medio. No se muestran más datos ya que no tiene parciales ni un mapa.

Los botones inferiores de guardar y borrar funcionan de la misma manera que los de la pantalla de visión de una actividad automática.

5.3.3.9. Pantalla de historial.

A esta pantalla se accederá a través de la barra de menú inferior de la aplicación. Mostrará un mapa que permitirá interactuar con los días de los que se guarda algún tipo de registro (desde pasos y calorías hasta actividades).

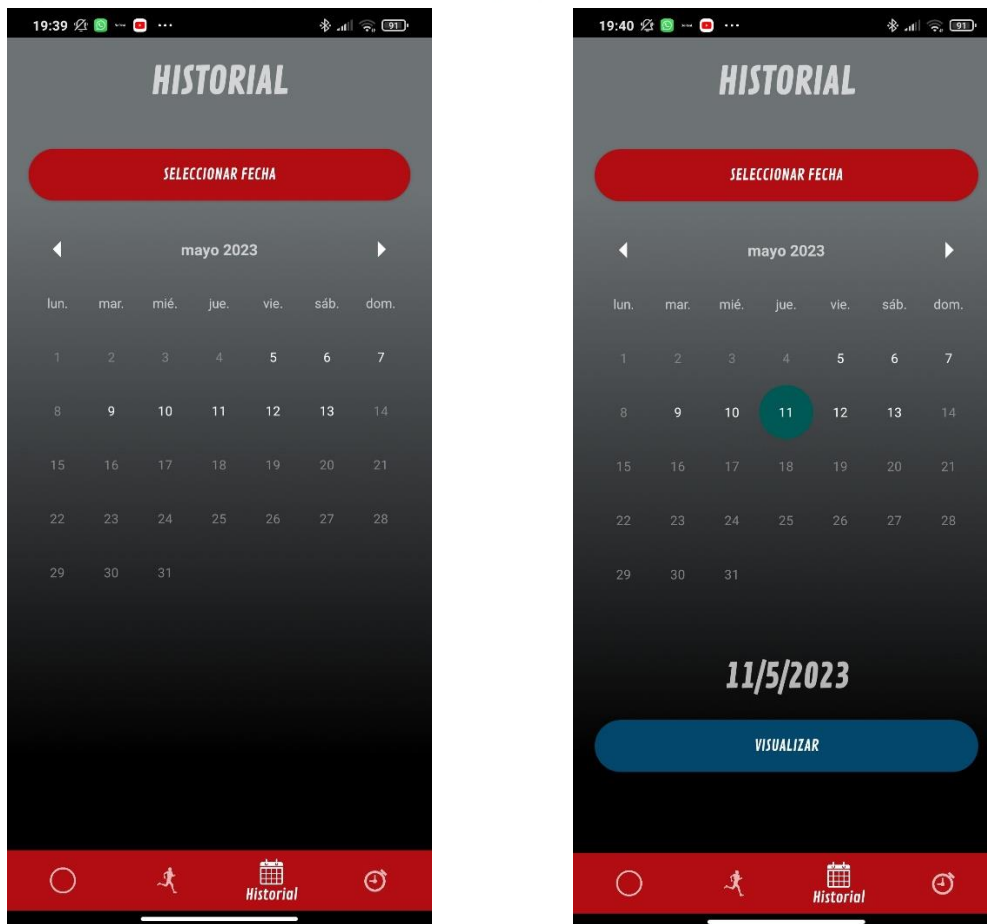


Ilustración 103. Pantalla de visión del historial.

Cómo se puede ver en la imagen superior, al seleccionar una fecha, se mostrará un botón que permite visualizar las métricas guardadas ese día.

En la parte superior de la pantalla al pulsar el botón de “SELECCIONAR FECHA”, se abrirá un dialog para seleccionar el año y la fecha que se quiere visualizar en el calendario.



Ilustración 104. Dialog selector de fecha.

5.3.3.10. Pantalla de visualización de un día histórico.

Esta pantalla será la encargada de mostrar los pasos, las calorías y las actividades (si las hubiera) que se realizaron en un día pasado. Se accede a ella a través del botón “VISUALIZAR” que se muestra al seleccionar una fecha en la pantalla de calendario.

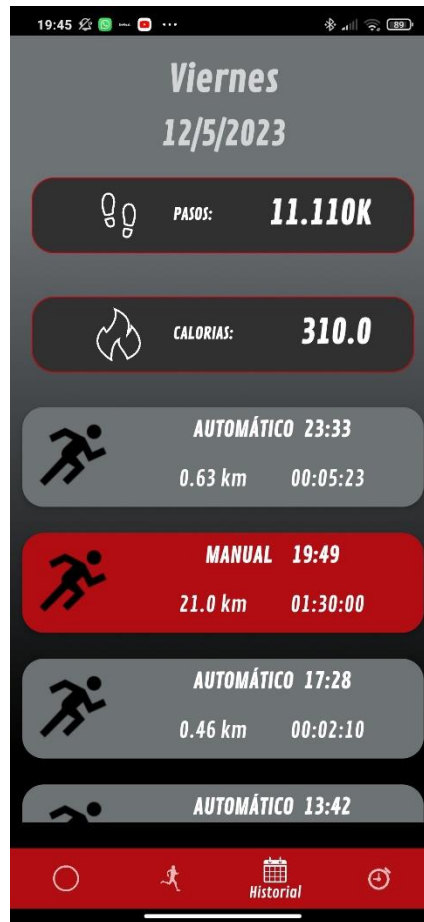


Ilustración 105. Pantalla de visualización de un día histórico.

Como se puede ver, la apariencia es muy parecida a la de la ventana principal, solo que no muestra ni el peso, ni la altura, ni la foto del usuario. Los elementos de las actividades son interactivos al igual que los de la ventana principal, y al pulsar sobre ellos mostrarán la ventana de visualización de esa actividad.

5.3.3.11. Pantalla de predicciones.

Esta pantalla será la encargada de mostrar las predicciones de tiempo que realizará el usuario en las distancias de 1km, 10km, 21km y maratón. Se accede a ella a través de la barra de menú ubicada en la parte inferior de la aplicación, pulsando sobre la última opción, llamada “PREDICCIONES”.



Ilustración 106. Pantalla de predicciones.

Como se puede ver en la imagen superior, si el usuario no cumple las condiciones necesarias para que el sistema pueda realizar las predicciones se le mostrará un mensaje explicativo.

En el momento que las cumpla, la pantalla cambiará y mostrará los 4 campos con los datos.



Ilustración 107. Pantalla de predicciones con datos.

5.3.3.12. Carga de datos.

Finalmente, es importante destacar, que para todas las operaciones que tardan un cierto tiempo en ejecutarse, se mostrará un elemento de carga. Su función, además de ser un complemento estético es la de notificar al usuario que su solicitud está siendo procesada.

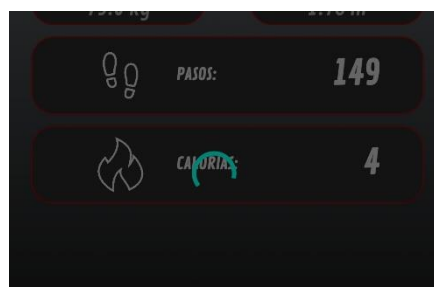


Ilustración 108. Carga de datos.

5.4. DSI 6: DISEÑO FÍSICO DE DATOS

En esta sección se describen los temas relativos al diseño físico de los datos, su ubicación y la manera de interactuar con ellos.

5.4.1. Descripción del SGBD usado.

Se ha optado por usar un SGBD totalmente online. No existe base de datos local. Para la implementación de la base de datos se ha usado Firebase Database con el objetivo de almacenar documentos en formato JSON y Firebase Storage para guardar imágenes.

En Firebase Database se guardan todos los datos de la aplicación:

- Datos de los usuarios.
- Registro de la última conexión de los usuarios.
- Pasos y calorías realizadas por cada usuario en cada día.
- Actividades tanto manuales como automáticas.
- Los 3 kilómetros más rápidos realizados por el usuario en distancias iguales o superiores a 3 kilómetros.
- Última conexión de cada usuario.

En Firebase Storage se guardarán las imágenes de perfil de cada usuario, identificadas con su nombre de usuario (que es único).

Usando esta base de datos en la nube, nos aseguramos de que cualquier usuario pueda acceder a sus datos históricos o diarios desde el dispositivo que sea.

5.4.2. Documentos

Firebase usa documentos JSON para almacenar la información.

Los datos de la aplicación están organizados en dos grandes colecciones:

- Colección de usuarios: Formada por todos los usuarios del sistema.
- Colección de registro: Guarda el último día del que se tiene registro del usuario en la aplicación.

El id de los documentos de ambas colecciones son los nombres de usuario

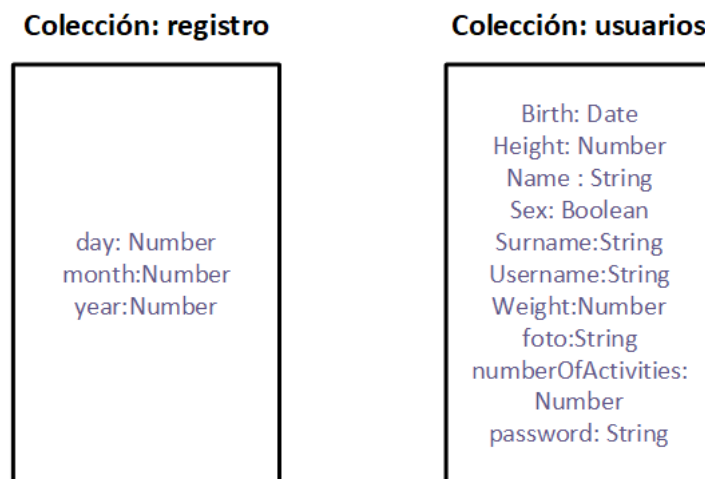


Ilustración 109. Documentos base de datos.

La colección de usuarios además está formada de otras colecciones para permitir guardar los datos diarios de cada usuario, siguiendo el siguiente esquema:

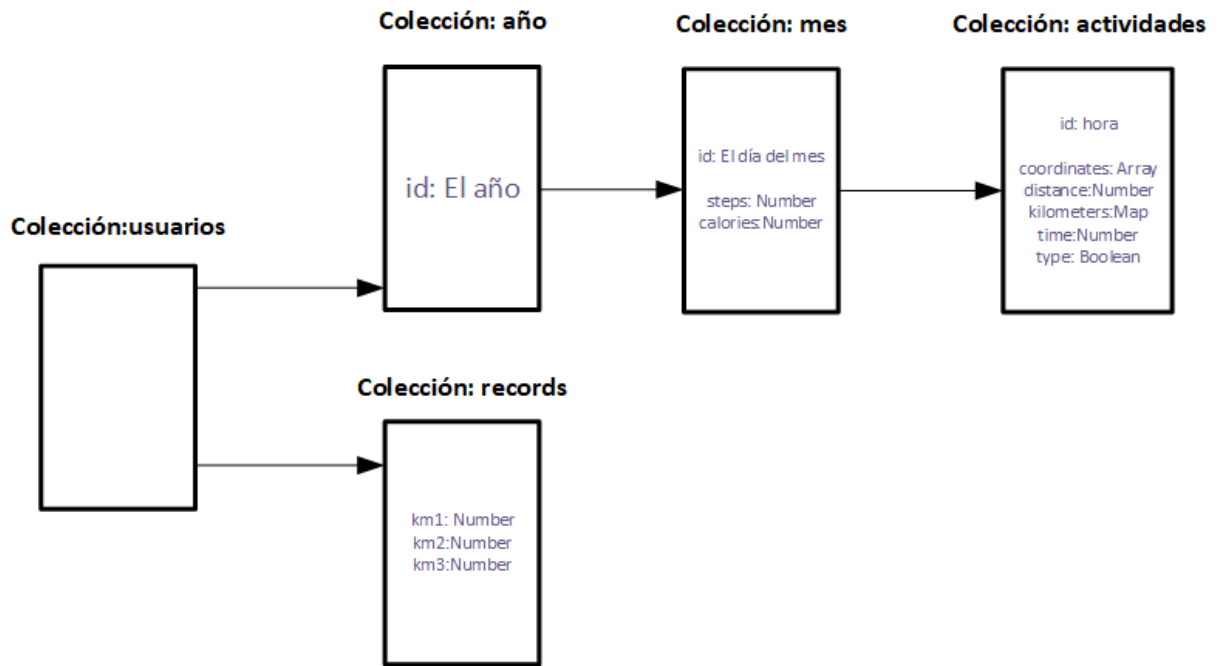


Ilustración 110. Diagrama base de datos.

Por tanto, para acceder a los detalles de un día concreto se hará de la siguiente manera:

usuarios/nombreUsuario/año/añoActual/mesActual/díaDeseado/actividades/horaDeseada/

Por ejemplo, así se accedería a los detalles de una actividad realizada el 13 de mayo de 2023 a las 17:23:05:

usuarios/nombreUsuario/año/2023/5/13/actividades/17:23:05

La colección records, guarda los 3 kilómetros más rápidos realizados por el usuario.

5.5. DSI 10: ESPECIFICACIÓN TÉCNICA DEL PLAN DE PRUEBAS

Para la realización de las pruebas de la aplicación se utilizará un ordenador portátil y un dispositivo móvil con Android.

Las características del dispositivo móvil son:

- **Modelo:** Xiaomi MI 9
- **CPU:** Octa-core Max 2.84GHz
- **RAM:** 6GB

- **Disco:** 64GB
- **Versión de Android:** 10

Por otra parte, las características técnicas del portátil son:

- **Modelo:** Hp Pavilionx360
- **CPU:** Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz
- **RAM:** 8GB
- **Disco:** Samsung SSD 840 EVO 500GB
- **Sistema Operativo:** Windows 10 Home.

Las herramientas que se van a usar para la realización de las pruebas son:

- **Pruebas unitarias:** JUnity.
- **Pruebas de integración:** Espresso.
- **Pruebas de usabilidad:** Cuestionarios.

5.5.1. Pruebas unitarias

El cometido de estas pruebas es analizar el correcto funcionamiento de la lógica del sistema.

Se han realizado pruebas de la lógica de las funcionalidades claves de cada elemento del sistema.

5.5.1.1. Registro de usuario

Registro de usuario	
Descripción	Resultado esperado
Prueba de contraseña. Se probarán diferentes contraseñas para registrarse.	Las contraseñas que cumplan las condiciones devolverán true, las que no, devolverán false.
Tiene texto. Método que prueba la comprobación de si tiene texto un editText.	Los elementos que no tengan texto o contengan texto vacío devolverán false. Los que contengan texto, devolverán true.
Comprobar peso. Método que comprueba los posibles valores de peso que puede introducir el usuario en la pantalla de registro.	Los pesos que no satisfagan las condiciones o que sean incorrectos devolverán false. Los pesos correctos devolverán true.
Comprobar altura. Método que comprueba los posibles valores de altura que puede introducir el usuario en la pantalla de registro.	Las alturas con formato correcto devolverán true. Las que tengan un formato incorrecto, devolverán false.
Registrar usuario. Registra a un usuario en el sistema, comprobando si el nombre de usuario ya está en el sistema.	El sistema registra al usuario, actualizando la base de datos y dejando la sesión iniciada.

Tabla 115. Pruebas unitarias registro de usuario.

5.5.1.2. Inicio de sesión

Inicio de sesión	
Descripción	Resultado esperado
Comprobar contraseña. Comprueba que la contraseña que se ha introducido para un usuario en concreto se corresponde con la que está codificada en la base de datos de ese usuario.	Codificará la contraseña introducida y la comparará con la que se introduce. Devolverá true si es la misma y false si no son iguales.

Tabla 116. Pruebas unitarias inicio de sesión.

5.5.1.3. Actividad principal

Actividad principal	
Descripción	Resultado esperado
Cerrar sesión. Se comprueba el funcionamiento del método de cerrar sesión.	En la base de datos se almacenan los pasos que realizó el usuario hasta ese momento.
Registro de la actividad del usuario. Comprueba que al iniciarse la actividad principal se actualice en la base de datos la última conexión del usuario.	La última conexión del usuario se actualizará en la base de datos.
Registro de la actividad con la aplicación iniciada. Comprueba que si se inicia la actividad un día más tarde (con la sesión ya iniciada). El registro de última conexión del usuario se actualizará.	La última conexión del usuario se actualizará a la del día actual.

Tabla 117. Pruebas unitarias actividad principal.

5.5.1.4. Grabación con los sensores del dispositivo

Grabación con los sensores del dispositivo	
Descripción	Resultado esperado
Añadir el kilómetro restante. Método que comprueba que cuando se realice una actividad de un número no redondo de kilómetros, se tratarán los últimos metros como un kilómetro más. Este tests es para distancias menores a un kilómetro (Por ejemplo 0.7m)	El elemento que devuelve el método contiene los últimos metros y el tiempo que se tardaron en realizar.
Añadir el kilómetro restante más de un kilómetro. Este método hace la misma comprobación que el anterior, pero en el caso de que la actividad sea mayor de 1 km. (Por ejemplo 1.3km)	El elemento que devuelve el método contiene los últimos metros y el tiempo que se tardaron en realizar.

Tabla 118. Pruebas unitarias grabación con los sensores del dispositivo.

5.5.1.5. Grabación manual

Grabación manual

Descripción	Resultado esperado
Guardado de actividad. Se guarda una actividad de una distancia y un tiempo concreto.	La actividad se guarda en la base de datos del usuario y se calculan el ritmo y la velocidad.

Tabla 119. Pruebas unitarias grabación manual.

5.5.1.6. Fragmento con información del usuario

MainWindowFragment	
Descripción	Resultado esperado
Conseguir información del usuario. Comprueba que, al crearse el fragmento, se consigue un elemento con toda la información del usuario.	Los campos del usuario están completos y son correctos.
Conseguir actividades de un usuario sin actividades. Se comprueba que al iniciarse el fragmento se consiga una lista vacía para los usuarios que no tienen actividades guardadas ese día.	El sistema devuelve una lista sin elementos.
Conseguir actividades de un usuario con actividades. Se comprueba que al iniciarse el fragmento se consiga una lista con las actividades que ha realizado el usuario ese día.	Se devuelve una lista con varias actividades.

Tabla 120. Pruebas unitarias fragmento con información del usuario.

5.5.1.7. Fragmento del historial

HistoryFragment	
Descripción	Resultado esperado
Desactivar días sin registros pasados. Comprueba el funcionamiento del método desactivar días cuando el usuario no tiene registrado ningún dato ese mes.	El método devuelve una lista vacía. Lo que indica que se desactivaran todos los días del mes en el calendario.
Desactivar días. Comprueba el funcionamiento del método desactivar días.	El método devuelve una lista con los días de los que se guarda registro y que no se van a desactivar en el calendario.
Desactivar días teniendo en cuenta el actual. Este método comprueba que el día actual también aparecerá desactivado para el usuario.	El método devuelve una lista con los días de los que se guarda registro, esta lista no contiene el día actual.

Tabla 121. Pruebas unitarias fragmento del historial.

5.5.1.8. Fragmento de predicciones

PredictionsFragment	
Descripción	Resultado esperado

Conseguir predicciones sin 3 actividades de mínimo 3 kilómetros. Testea el método que comprueba si se cumplen las condiciones para realizar predicciones en el caso de que no se cumplan.	El método devolverá false.
Conseguir predicciones. Testea el método que comprueba si se cumplen las condiciones para realizar las predicciones en el caso que se satisfacen.	El método tras hacer una comprobación en la base de datos devolverá true.

Tabla 122. Pruebas unitarias fragmento de predicciones.

5.5.1.9. Detalles de una actividad automática

Train Details	
Descripción	Resultado esperado
Obtención de datos de una actividad de un único punto. Comprueba que al iniciar la actividad se carguen todos los datos de la actividad automática. En este caso se prueba con una actividad de la que sólo se ha guardado un punto geográfico.	Se obtendrá un objeto con todos los datos correctos de la actividad.
Obtención de datos de una actividad con muchos puntos. Comprueba que al iniciar la actividad se carguen todos los datos de la actividad automática. En este caso se prueba con una actividad de la que se han guardado varios puntos geográficos.	Se obtendrá un objeto con todos los datos correctos de la actividad.
Obtención de datos de una actividad con paradas. Comprueba que al iniciar la actividad se carguen todos los datos de la actividad automática. En este caso se prueba con una actividad de la que se han guardado varios puntos geográficos y además se han realizado paradas y reanudaciones.	Se obtendrá un objeto con todos los datos correctos de la actividad.
Separación si es necesaria. Este método comprueba que una actividad con paradas es tratada correctamente para dividirla en las secciones de puntos geográficos.	De la lista de puntos con pausas, se obtendrán varias listas de puntos.

Tabla 123. Pruebas unitarias detalles de una actividad automática.

5.5.1.10. Detalles de una actividad manual

Train Details Manual Activity	
Descripción	Resultado esperado
Obtención de datos de una actividad manual. Testea que al iniciarse la actividad se obtengan correctamente los datos de esta.	Se obtendrá un objeto con todos los datos correctos de la actividad.

Obtención de datos de una actividad de una hora específica. Testea que, al iniciarse el muestreo de datos de una actividad en una hora específica, se obtengan sus datos correctamente.	Se obtendrá un objeto con todos los datos correctos de la actividad.
---	--

Tabla 124. Pruebas unitarias detalles de una actividad manual.

5.5.1.11. Métodos utils

Utils	
Descripción	Resultado esperado
Conversión de formato de segundos. Testea el método que convierte los segundos en un formato de tiempo.	Se obtendrá una cadena con el formato esperado (similar al de un cronómetro).
Conversión de formato de segundos a min/km. Testea el método que convierte los segundos en un formato de min/km.	Se obtendrá una cadena con el formato esperado.
Conversión a min/km. Testea el funcionamiento del método que convierte la velocidad a una cadena con formato min/km.	Se obtendrá una cadena con el formato esperado.

Tabla 125. Pruebas unitarias utils.

5.5.2. Pruebas de integración

Estas pruebas comprueban el correcto funcionamiento del sistema en su conjunto.

Al igual que en las unitarias, se han diseñado tests para las funcionalidades principales de cada subsistema.

5.5.2.1. Registro de usuario

Registro de usuario	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están.
Tiene texto. Comprueba que se muestre un mensaje de error si se intenta registrar un usuario sin haber introducido texto en todos los campos.	Se mostrará un mensaje de error específico si se registra un usuario sin haber rellenado todos los campos.
Comprobación de peso. Comprueba que se notifica al usuario si introduce erróneamente el formato del peso.	Se mostrará un mensaje de error cuando el formato sea incorrecto, y no se mostrará cuando sea válido.
Comprobación de altura. Comprueba que se notifica al usuario si introduce erróneamente el formato de la altura.	Se mostrará un mensaje de error cuando el formato sea incorrecto, y no se mostrará cuando sea válido.
Comprobación de usuario identificado. Comprueba que se notifica al usuario si	Se mostrará un mensaje de error específico y el campo de nombre de usuario quedará

intenta registrarse con un nombre de usuario que ya está guardado en el sistema.	vacío. Cuando el formato sea correcto se registrará al usuario.
Comprobación de contraseña. Comprueba que se notifica al usuario cuando el formato de la contraseña no es el correcto, ya sea por longitud o falta de caracteres.	Se mostrará un mensaje de error específico y el campo de contraseña quedará vacío. Cuando el formato sea correcto se registrará al usuario.
Mover a inicio de sesión. Se comprueba el funcionamiento del botón que dirige al usuario a la pantalla de inicio de sesión.	Si el usuario pulsa el botón se le dirigirá a la pantalla de inicio de sesión. Al volver a la de registro, todos los campos estarán vacíos.

Tabla 126. Pruebas de integración registro de usuario.

5.5.2.2. Inicio de sesión

Inicio de sesión	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están.
Tiene texto. Comprueba que se muestre un mensaje de error si se intenta iniciar sesión sin haber introducido texto en todos los campos.	Se mostrará un mensaje de error específico si se inicia sesión sin haber rellenado todos los campos.
Comprobación de usuario existente. Comprueba que se muestra un error si el usuario intenta iniciar sesión con un nombre de usuario que no existe en el sistema.	Se mostrará un mensaje de error. En el momento que se introduzca un nombre correcto, se iniciará la sesión.
Comprobación de contraseña. Comprueba que se notifique al usuario si se intenta iniciar sesión con una contraseña que no se corresponde a la que se tiene guardada en el sistema para ese usuario.	Se mostrará un mensaje de error. En el momento en el que se introduzca la contraseña correcta, se iniciará la sesión.
Mover a registro. Comprueba el funcionamiento del botón que dirige al usuario a la pantalla de registro.	Al pulsar el botón se redirigirá al usuario a la pantalla de registro, si se vuelve a la pantalla de inicio de sesión, todos los campos estarán vacíos.

Tabla 127. Pruebas de integración inicio de sesión.

5.5.2.3. Actividad principal

MainActivity	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están. Los datos del usuario aparecen correctamente tras el tiempo de carga.
Cambiar de fragmento. Comprueba el funcionamiento del menú inferior que cambia el fragmento que se muestra en la actividad principal.	Al pulsar los botones del menú inferior se cambiará al fragmento esperado. Si se pulsa el botón atrás se volverá al fragmento anterior.

Tabla 128. Pruebas de integración actividad principal.

5.5.2.4. Grabar actividad con los sensores del dispositivo

RecordActivity	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están.
Comenzar actividad. Comprueba el funcionamiento del inicio y pausado de la actividad.	La actividad se inicia y se pausa según lo pulsado en los botones de la pantalla. Los botones se mostrarán o no dependiendo de si la actividad está iniciada o pausada.
Pulsar hacia atrás con la actividad no iniciada. Comprueba el comportamiento de si se pulsa hacia atrás cuando la actividad aún no ha comenzado.	Se muestra la pantalla principal y no se ha guardado ninguna actividad.
Pulsar hacia atrás con la actividad iniciada. Comprueba el comportamiento de si se pulsa hacia atrás cuando la actividad aún ha comenzado.	Se muestra la pantalla principal y no se ha guardado ninguna actividad.
Pulsar hacia atrás con la actividad pausada. Comprueba el comportamiento de si se pulsa hacia atrás cuando la actividad está pausada.	Se muestra la pantalla principal y no se ha guardado ninguna actividad.
Guardar actividad. Comprueba el funcionamiento de la finalización de la actividad.	Se para y guarda la actividad y se muestran los detalles de esta.

Tabla 129. Pruebas de integración grabar actividad con los sensores del dispositivo.

5.5.2.5. Grabar actividad manualmente

ManualRecordActivity	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están.
Pulsar hacia atrás. Comprueba el funcionamiento de pulsar hacia atrás.	Se redirigirá al usuario a la pantalla principal y no se habrá guardado ninguna actividad.
Activar botón. Comprueba si se activa correctamente el botón de guardar al interactuar con los elementos de la pantalla.	El botón de guardar actividad se activará cuando los valores de distancia y tiempo sean distintos de 0.
No activar botón. Comprueba que, aunque se de valor al tiempo, si la distancia es 0, el botón no se activa.	El botón de guardar permanecerá desactivado.
Guardar actividad. Comprueba el funcionamiento del guardado de una actividad manual.	Se guarda la actividad y se dirige al usuario a los detalles de esta.

Tabla 130. Pruebas de integración grabar actividad manualmente.

5.5.2.6. Fragmento con información del usuario

MainWindowFragment	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están.
Cambiar peso. Comprueba el funcionamiento de cambiar el peso del usuario.	El peso se actualizará en la interfaz gráfica y en la base de datos.
Cambiar altura. Comprueba el funcionamiento de cambiar la altura del usuario.	La altura se actualizará en la interfaz gráfica y en la base de datos.
Mostrar actividad. Comprueba que al pulsar sobre un elemento actividad del fragmento principal se muestren los detalles de esta.	Se muestran los detalles de la actividad.
Cerrar sesión. Comprueba el funcionamiento de cerrar la sesión.	Se cierra la sesión, se actualiza el registro en la base de datos.

Tabla 131. Pruebas de integración fragmento con información del usuario.

5.5.2.7. Fragmento del historial

HistoryFragment	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles. También se comprueba que aparece activado únicamente el día del que se guarda registro.	Los elementos que deben de estar visibles lo están.
Cambiar mes. Testea el funcionamiento de del cambio del mes del calendario.	El mes se cambia correctamente. Cambian los días activados.
Ver un día histórico. Comprueba el funcionamiento de ver un día histórico.	Se muestran los detalles del día seleccionado.

Tabla 132. Pruebas de integración fragmento del historial.

5.5.2.8. Fragmento de un día histórico

HistoricalDayFragment	
Descripción	Resultado esperado
Todo en su lugar. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Los elementos que deben de estar visibles lo están.
Abrir actividad. Comprueba el funcionamiento de ver los detalles de una actividad que se realizó en un día pasado.	Se muestran los detalles de esa actividad.
Pulsar hacia atrás. Comprueba el funcionamiento de pulsar el botón de ir hacia atrás.	Se redirige al usuario a la pantalla de visión del historial.

Tabla 133. Pruebas de integración fragmento de un día histórico.

5.5.2.9. Fragmento de predicciones

PredictionsFragment	
Descripción	Resultado esperado
Todo en su lugar cuando no hay suficientes actividades para realizar predicciones. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Se muestran todos los elementos, entre ellos el mensaje explicativo de que no se pueden realizar aún predicciones.
Todo en su lugar cuando hay suficientes actividades para realizar predicciones. Comprueba que al iniciarse la actividad todos los elementos estén posicionados correctamente y visibles.	Se muestran todos los elementos, entre ellos las predicciones.

Tabla 134. Pruebas de integración fragmento de predicciones.

5.5.2.10. Detalles de una actividad automática

Train Details	
Descripción	Resultado esperado
Detalles de una actividad de un solo punto. Se comprueba la muestra de detalles de una actividad que solo tiene un punto geográfico y en la que el usuario no se ha desplazado.	Se muestran todos los elementos correctamente.
Detalles de una actividad de varios puntos geográficos. Se comprueba la muestra de detalles de una actividad que tiene varios puntos geográficos.	Se muestran todos los elementos correctamente.
Detalles de una actividad de varios puntos geográficos y con paradas. Se comprueba la muestra de detalles de una actividad que tiene varios puntos geográficos y paradas.	Se muestran todos los elementos correctamente.
Eliminar una actividad. Comprueba el proceso de eliminación de una actividad.	La actividad se elimina y se redirige al usuario a la pantalla principal, donde no se muestra la actividad.
Guardar actividad. Comprueba el proceso de guardar la actividad.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad.
Pulsar el botón hacia atrás. Comprueba el proceso que se ejecuta tras pulsar el botón de atrás.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad.

Tabla 135. Pruebas de integración detalles de una actividad automática.

5.5.2.11. Detalles de una actividad manual

Train Details Manual Activity	
Descripción	Resultado esperado

Todo en su lugar. Se comprueba que todos los elementos de la actividad están en su lugar y correctamente posicionados.	Todos los elementos se muestran correctamente.
Eliminar una actividad. Comprueba el proceso de eliminación de una actividad.	La actividad se elimina y se redirige al usuario a la pantalla principal, donde no se muestra la actividad.
Guardar actividad. Comprueba el proceso de guardar la actividad.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad.
Pulsar el botón hacia atrás. Comprueba el proceso que se ejecuta tras pulsar el botón de atrás.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad.

Tabla 136. Pruebas de integración detalles de una actividad manual.

5.5.2.12. Pruebas adicionales

Para testear el correcto funcionamiento de los apartados que tienen que ver con la toma de métricas, se van a realizar ciertas pruebas adicionales. No se tratan de pruebas generadas por código, se basan en comparar las métricas recogidas por la aplicación con las tomadas por otra aplicación, sistema o de las reales.

Es importante destacar que este sistema se trata de un prototipo y que no pretende competir contra aplicaciones o empresas que tienen más medios y profesionales. Con estas pruebas lo que se pretende comprobar es si las métricas son coherentes o no. También es necesario mencionar que se pueden encontrar diferencias entre las mediciones que toman sistemas profesionales como Garmin o Strava.

5.5.2.11.1. Prueba de pasos

Esta prueba pretende conocer si los pasos que mide la aplicación son coherentes respecto a los que de verdad da el usuario. Para esta prueba se van a dar 300 pasos exactos para saber la efectividad del sistema.

5.5.2.11.2. Prueba de grabado de la actividad

Con esta prueba se pretende conocer la efectividad de la aplicación a la hora de grabar actividades con los sensores del dispositivo. Para realizarlas, se guardarán una serie de actividades con el Reloj GPS Gamin Forerunner 245 (de la marca Garmin) y con la aplicación desarrollada en este trabajo.

Se compararán las actividades en los aspectos de:

- Mapa mostrado
- Tiempo tomado
- Distancia tomada
- Parciales

5.5.2.11.3. Prueba de predicciones

Para conocer la fiabilidad de las predicciones se va a pedir a 3 usuarios distintos que introduzcan en el sistema 3 actividades de más de 3 kilómetros con sus tiempos reales. Posteriormente se les pedirá que notifiquen su grado de satisfacción en base a la siguiente escala.

Grado de satisfacción con las predicciones.				
¿Cómo de factible es realizar esos tiempos?				
0. IMPOSIBLE	1. POCO PROBABLE	2. POSIBLE	3. PROBABLE	4. MUY PROBABLE

Tabla 137. Grado de satisfacción con las predicciones

5.5.3. Pruebas de usabilidad

El objetivo de estas pruebas es testear y mejorar la usabilidad del prototipo. Para ello, se usarán usuarios de prueba que proporcionarán feedback para mejorar el prototipo.

Las pruebas se realizarán de la siguiente manera:

1. Se le dejará el prototipo al usuario y un cuestionario a rellenar.
2. Se le propondrán una serie de actividades guiadas a realizar.
3. Si lo desea puede añadir sugerencias o anotaciones.
4. A partir de los resultados se tomarán medidas.

Se usarán 4 usuarios de diferentes grupos de edad para probar la aplicación.

5.5.3.1. Actividades guiadas

A continuación, se detallan las actividades guiadas que se le propondrán a los usuarios. Al terminar cada una se anotará si fue posible realizarlas o no.

Actividades guiadas
1. Regístrate en el sistema con tus credenciales.
2. Modifica tu foto de perfil.
3. Modifica tu altura 1 cm.
4. Realiza una actividad automática (la distancia o el tiempo es indiferente).
5. Borra la actividad.
6. Visualiza los datos de un día pasado.
7. Graba 3 actividades manuales de más de 3 kilómetros.
8. Visualiza las predicciones.
9. Cierra sesión.

Tabla 138. Actividades guiadas.

5.5.3.2. Cuestionario.

Se realizará a cada usuario un cuestionario dividido en las siguientes partes:

- Preguntas para recoger información del usuario.
- Preguntas y anotaciones sobre las actividades guiadas.
- Cuestionario de usabilidad.

5.5.3.2.1. Preguntas sobre el usuario.

Se realizarán las siguientes preguntas para contextualizar al usuario de prueba.

Preguntas referentes al usuario.	
¿Qué edad tienes?	
<ul style="list-style-type: none">• Entre 18 y 25 años.• Entre 26 y 40 años.• Entre 40 y 55 años.• Más de 55 años.	
¿Cuántas horas dedicas diariamente al uso del dispositivo móvil?	
<ul style="list-style-type: none">• Menos de una hora.• Entre 1 y 2 horas.• Más de 2 horas.	
¿Ha usado anteriormente aplicaciones para el seguimiento de métricas deportivas?	
<ul style="list-style-type: none">• Sí.• No.	
En caso afirmativo, por favor indique cuales:	
¿Qué busca en una aplicación de estas características?	
<ul style="list-style-type: none">• Facilidad de uso.• Métricas muy precisas.• Velocidad de carga.• Funcionalidades innovadoras.	

Tabla 139. Preguntas referentes al usuario.

5.5.3.2.2. Preguntas y anotaciones sobre las actividades guiadas.

Para cada una de las actividades guiadas se responderá la siguiente pregunta.

¿Has sido capaz de realizar la actividad?
<ul style="list-style-type: none">• Sí.• No.

Tabla 140. Preguntas y anotaciones sobre las actividades guiadas.

Finalmente se podrán anotar los problemas encontrados.

5.5.3.2.3. Cuestionario de usabilidad

Evaluará el apartado visual y la usabilidad de la aplicación.

Preguntas de usabilidad.					
Valora la interfaz gráfica de la aplicación.					
0. Fea.	1.	2.	3.	4.	5. Muy bonita.
Valora la usabilidad de la aplicación.					

0. Difícil de usar	1.	2.	3.	4.	5. Muy sencilla de usar.
Valora la navegabilidad de la aplicación.					
0. Mala navegabilidad	1.	2.	3.	4.	5. Muy buena navegabilidad
Valora los tiempos de carga.					
0. Tiempos muy largos	1.	2.	3.	4.	5. Tiempos de carga inapreciables.
Valora los colores de la interfaz gráfica (múltiple elección).					
<ul style="list-style-type: none"> • No combinan. • Combinan perfectamente. • Diferencian correctamente las opciones negativas y positivas. • No diferencian las opciones positivas y negativas. • Son amigables a la vista. • Son monótonos. • Son correctos. 					

Tabla 141. Preguntas de usabilidad.

Finalmente, el usuario podrá introducir sus opiniones y propuestas sobre la aplicación.

5.5.3.3. Cuestionario para el responsable de pruebas

El responsable de pruebas irá rellenando el siguiente cuestionario mientras cada uno de los usuarios de prueba usan la aplicación.

Cuestionario para el responsable de pruebas		
	<i>Observaciones.</i>	<i>Problemas encontrados.</i>
Navegación por la interfaz.		
Grabación de actividades automáticas.		
Modificación de datos personales.		
Uso del histórico.		
Grabación de actividades manuales.		
Visualización y borrado de actividades.		
Visualización de predicciones.		

Tabla 142. Cuestionario para el responsable de pruebas.





CAPÍTULO 6: CONSTRUCCIÓN DEL SISTEMA DE INFORMACIÓN

FASE DE DESARROLLO

DSI

6.1. CSI 1: PREPARACIÓN DEL ENTORNO DE GENERACIÓN Y CONSTRUCCIÓN

En esta sección se tratan algunos aspectos relativos a la implementación del software desarrollado.

6.1.1. Estándares y normas seguidos

Durante el desarrollo del software se ha procurado seguir una serie de estándares y normas para facilitar y normalizar las fases de desarrollo e implantación.

6.1.1.1. *Java Code Conventions.*

Java Code Conventions [28] es un estándar usado para facilitar y normalizar la lectura y comprensión del software.

6.1.1.2. *Arquitectura MVC*

Modelo Vista Controlador [29] es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta, lo que ofrece beneficios diversos.



Ilustración 111. Arquitectura MVC.

6.1.2. Lenguajes empleados

En el siguiente apartado se listarán los lenguajes de programación usados en el desarrollo del software del sistema.

6.1.2.1. Java

Java [30] es un lenguaje de programación orientado a objetos. Se decidió usar este lenguaje porque es el que más se utilizó durante el desarrollo del grado por parte del usuario. Es un lenguaje sencillo e imperativo, que permite desarrollar aplicaciones en Android Studio. Es uno de los lenguajes más populares en la actualidad.



Ilustración 112. Java.

6.1.2.2. XML

Para el desarrollo de la interfaz gráfica se utilizó XML [31], o también conocido como lenguaje de marcado extensible. Permite definir y almacenar datos de forma compartible, permitiendo el intercambio de información entre sitios web, bases de datos y aplicaciones de terceros.

Es un lenguaje de marcado que proporciona reglas para definir cualquier dato.



Ilustración 113. XML.

6.1.3. Librerías y frameworks

A continuación, se especifican las librerías y frameworks usados en el proyecto.

6.1.3.1. Material Calendar View

Material-calendarView [32] es una librería de código abierto para Android que proporciona un calendario interactivo para aplicaciones. Se utilizó esta librería porque permite un gran grado de personalización, así como desactivar y activar días concretos.

6.1.3.2. Mapbox

Mapbox [9] es un proveedor de mapas en línea realizados por encargo. Proporciona varias APIs para agregar mapas a aplicaciones o sitios web, también permite personalizar estos mapas e interactuar con ellos. Se utiliza para el muestreo del mapa y de la línea de seguimiento del usuario.



Ilustración 114. Mapbox.

6.1.3.3. Image-cropper

Image-cropper [33] es una librería de código abierto que permite seleccionar y recortar imágenes del dispositivo o tomarlas directamente con la cámara de este. Es usada para la gestión de la imagen del usuario de la aplicación.

6.1.3.4. Espresso testing

Espresso [34] es un framework de testing open source lanzado por Google, proporciona una API para realizar pruebas de interfaz de usuario en aplicaciones Android. Todas las pruebas de integración se realizaron con esta API.



Ilustración 115. Espresso.

6.1.3.5. Fragment-testing

Fragment-testing [35] es una API que permite crear pruebas sobre fragmentos de Android Studio sin tener que instanciar una actividad que los contenga. Fue utilizada para crear los test unitarios de los fragmentos del sistema.

6.1.4. Herramientas y programas usados para el desarrollo

En este apartado se detallan las herramientas y los programas que se han usado en mayor o menor medida para el desarrollo de la aplicación.

6.1.4.1. Android Studio

Android Studio [12] es el IDE oficial de Android y que permite crear y compilar aplicaciones Android de una manera más rápida y cómoda en todos los dispositivos Android. Actualmente soporta los lenguajes Kotlin, Java y XML.

android studio



Ilustración 116. Android studio.

6.1.4.2. Cloud Firestore

Firebase Firestore o Cloud Firestore [36], es un producto de Google Cloud que consiste en una base de datos NoSQL flexible, escalable y en la nube. Mantiene los datos sincronizados entre apps cliente a través de objetos de escucha en tiempo real y ofrece soporte sin conexión para dispositivos móviles y la Web, por lo que se puede compilar apps con capacidad de respuesta que funcionan sin importar la latencia de la red ni la conectividad a Internet.

Firebase Firestore es utilizada para guardar todos los datos de clientes, actividades y registros de la aplicación.



Ilustración 117. Cloud Firestore.

6.1.4.3. Cloud Storage

Cloud Storage [37] es un servicio administrado para almacenar datos no estructurados. Almacena en la nube cualquier archivo, independiendo del tamaño, y permite recuperarlo en el momento en el que se necesite.

Cloud Storage es utilizada para guardar las imágenes de perfil de cada uno de los clientes de la aplicación.



Ilustración 118. Google Cloud Storage.

6.1.4.4. *Free Logo Maker*

Free Logo Maker [27] es un sitio web que a través de una serie de parámetros e inteligencia artificial te permite crear logos para una empresa, aplicación, u otros. El logo de la aplicación del sistema se creó usando esta herramienta.



Ilustración 119. Free LogoMaker.

6.1.5. Herramientas y programas usados para la documentación

Finalmente, en este apartado se listan todas las herramientas usadas para la documentación del sistema.

6.1.5.1. *Word*

Microsoft Word [38] es un software de tratamiento de textos creado por Microsoft. Es usada para realizar la documentación del sistema.



Ilustración 120. Word.

6.1.5.2. *Visio*

Visio [26] es una aplicación de diagramación y de gráficos vectoriales que forma parte de la suite ofimática de Microsoft Office. Fue utilizada para el desarrollo de todos los diagramas que aparecen en la documentación.



Ilustración 121. Visio.

6.1.5.3. Excel

Excel [39] es un programa que permite editar hojas de cálculo, desarrollada por Microsoft. Fue utilizada para la realización de algunas tablas y la elaboración del presupuesto en el apartado [3.1.5 Presupuesto inicial](#).



Ilustración 122. Excel.

6.1.5.4. Project

Microsoft Project [23] es un software de gestión de proyectos profesionales desarrollada por Microsoft que permite mantener un control sobre todos los procesos. Es usado en el apartado [Capítulo 3: Planificación del Sistema de Información](#).



Ilustración 123. Project.

6.1.5.5. Balsamiq

Balsamiq [40] es una herramienta para la creación de bocetos de interfaz de usuario. Fue utilizada en el apartado [4.8. ASI 8: Definición de interfaces de usuario](#).

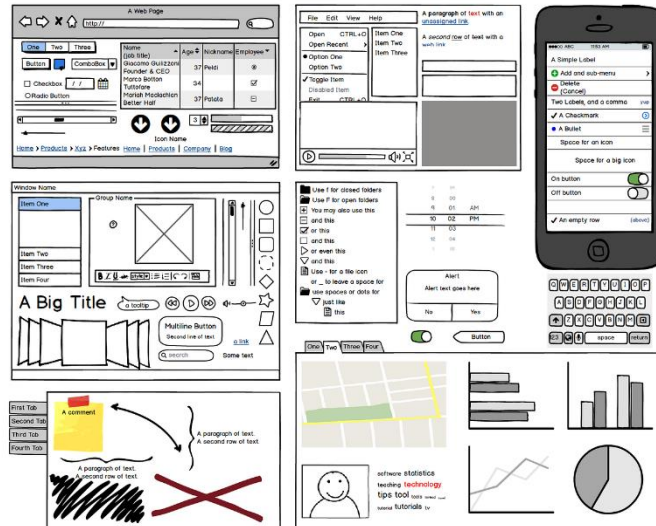


Ilustración 124. Balsamiq.

6.1.5.6. Google Forms

Google Forms es un software de administración de encuestas que se incluye como parte del conjunto gratuito de Google Docs Editors. Es utilizado para la realización de las pruebas de usabilidad en [5.3.3. Pruebas de usabilidad](#).



Google Forms

Ilustración 125. Google Forms.

6.2. CSI 2: GENERACIÓN DEL CÓDIGO DE LOS COMPONENTES Y PROCEDIMIENTOS

En el siguiente apartado se mostrarán algunos de los fragmentos de código más importantes de la aplicación. Se ha escogido código perteneciente al subsistema de grabado de actividades con los sensores del dispositivo y al subsistema de conteo de pasos y caloría. Junto a cada fragmento habrá una descripción.

6.2.1. Grabado de actividades con los sensores del dispositivo.

El sistema pedirá al usuario los permisos de ubicación, para poder tomar los puntos en primer o segundo plano. Se comprueba que permisos hay que pedir:

```
/**
 * Devuelve el estado de las peticiones necesitadas.
 */
private boolean checkPermissions() {
    return PackageManager.PERMISSION_GRANTED == ActivityCompat.checkSelfPermission(context: this,
        Manifest.permission.ACCESS_FINE_LOCATION);
}

private void requestPermissions() {
    Log.i(TAG, msg: "Requesting permission");
    ActivityCompat.requestPermissions(activity: RecordActivity.this,
        new String[]{Manifest.permission.ACCESS_FINE_LOCATION},
        REQUEST_PERMISSIONS_REQUEST_CODE);
    progress.dismiss();
}
```

Ilustración 126. Grabado de actividades con los sensores del dispositivo: Permisos.

Tras esto, cuando la petición sea realizada, se ejecutará un Callback que recibirá los resultados de esta.

```
/**
 * Callback recibida cuando la petición de localización es completada.
 */
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
    @NonNull int[] grantResults) {
    Log.i(TAG, msg: "onRequestPermissionResult");
    if (requestCode == REQUEST_PERMISSIONS_REQUEST_CODE) {
        if (grantResults.length <= 0) {
            // Si la interacción con el usuario es cancelada.
            Log.i(TAG, msg: "User interaction was cancelled.");
        } else if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            //Aquí se inicia el servicio en segundo plano.
            mService.requestLocationUpdates(time);
        } else {
            //Si no se dan permisos se cierra la actividad
            finish();
            startMainActivity();
        }
    }
}
```

Ilustración 127. Grabado de actividades con los sensores del dispositivo: Callback.

Si el usuario concede permisos, comenzará el servicio de toma de ubicación en segundo plano. En cambio, si la aplicación se cierra o si los deniega, se cerrará actividad y se redirigirá al usuario a la pantalla principal.

Una vez iniciado el servicio, el método onNewLocation se ejecutará cuando el sensor GPS capte una nueva ubicación.

```
private void onNewLocation(Location location) {
    Log.i(TAG, msg: "New location: " + location);
    mLocation = location;
    currentLocation = location;
    System.out.println("PREVIOUS LOCATION: "+previousLocation);
    System.out.println(currentLocation);
    if (stopped){
        previousLocation=location;
        stopped=false;
    }
    if (currentLocation==null || previousLocation==null){
        //En este momento se ha perdido la ubicación.
        previousLocation=currentLocation;
        progress.show();
        return;
    }
    //Aquí se ha encontrado la ubicación
    if(progress.isShowing())
        progress.dismiss();
}
```

Ilustración 128. Grabado de actividades con los sensores del dispositivo: onNewLocation.

En la primera parte del método, mLocation es el atributo usado para guardar los puntos del sistema, mientras que previousLocation y currentLocation se usarán para calcular la distancia recorrida. Si el sistema está pausado, no se sumará distancia y si no se encuentran ubicaciones, se mostrará un mensaje informando que se están buscando.

```
double d = distancia(previousLocation, currentLocation);
if (firstLocation && d>0.07){
    //Por si hay un desfase al comenzar la actividad.
    d = 0;
    firstLocation=false;
}
kilometers += d; //Suma la distancia
saveRoute(location, kilometers); //Comprueba si hay que actualizar los parciales y los puntos geográficos
sendDataToActivity(kilometers); //Manda los datos a la actividad
double speed = velocidad(location);
sendSpeedToActivity(speed); //Manda la velocidad a la actividad

// Envía localizaciones nuevas
Intent intent = new Intent(ACTION_BROADCAST);
intent.putExtra(EXTRA_LOCATION, location);
LocalBroadcastManager.getInstance(getApplicationContext()).sendBroadcast(intent);

// Actualiza la notificación.
if (serviceIsRunningInForeground( context: this)) {
    mNotificationManager.notify(NOTIFICATION_ID, getNotification());
}
previousLocation=location;
```

Ilustración 129. Grabado de las actividades con los sensores del dispositivo: onNewLocation 2.

En la siguiente parte del método se hace una comprobación para prever que en la primera ubicación haya desfase.

Seguidamente se calcula la distancia entre las dos ubicaciones, se envían los datos a la actividad a través de Broadcast receivers, se actualiza la notificación si el servicio está en segundo plano y se ejecuta el método `saveRoute`.

```
private void saveRoute(Location location, double km) {
    int decimas = getDecimas(km);
    if (decimas - decimasAntiguas >= 1) { //Aquí se comprueba si se han recorrido 100 metros
        decimasAntiguas = decimas;
        route.add(new PointDTO(location.getLatitude(), location.getLongitude()));
    } else if (decimasAntiguas > decimas) { //Aquí es que se ha recorrido un kilómetro
        double time = timeTask.getTime() - timeInitial;
        timeInitial = timeTask.getTime();
        //Guardado del kilómetro
        velocityPerKilometer.put(getKilometer(km) + "", time);
        decimasAntiguas = decimas;
    }
}
```

Ilustración 130. Grabado de las actividades con los sensores del dispositivo: `saveRoute`.

El método guardará un nuevo punto cuando se hayan recorrido 100 metros y actualizará los parciales cuando se comience un nuevo kilómetro.

6.2.2. Toma de pasos y cálculo de calorías

La toma de pasos también se realizará en un servicio en segundo plano, los pasos se calcularán a través del sensor de acelerómetro del dispositivo, ya que, aunque hay dispositivos que cuentan con sensor de pasos, no todos lo tienen. La petición de permisos de uso del sensor es prácticamente igual que la de localización, una vez conseguidos los permisos, se comienza la actividad en segundo plano.

Cada vez que el sensor detecte un movimiento se ejecutará el siguiente método:

```
@Override
public void onSensorChanged(SensorEvent event) {
    if(event!=null){
        //Se accede al sensor de acelerómetro y se consiguen los movimientos en los 3 ejes
        float x_acc = event.values[0];
        float y_acc = event.values[1];
        float z_acc = event.values[2];

        double magnitud = Math.sqrt(x_acc*x_acc + y_acc*y_acc * z_acc*z_acc);
        double magnitudDelta = magnitud -magnitudPrevious;
        magnitudPrevious = magnitud;
        //Si ha habido un cambio superior a STEP_CONSTANT, se suma un paso
        if (magnitudDelta>STEPS_CONSTANT){
            steps++;
            SharedPreferences.Editor editor =MainActivity.sharedPreferences.edit();
            editor.putInt("stepCount", steps);
            editor.apply();
        }
    }
}
```

Ilustración 131. Toma de pasos y calorías: onSensorChanged.

Tomará las componentes x,y,z a partir del sensor y hará un cálculo para obtener el grado de movimiento del dispositivo. Si ese valor es superior a un umbral, se contará como un paso, que se actualizará en el fichero SharedPreferences.

Los pasos serán enviados al usuario a través de un BroadcastReceiver, y también se actualizarán en una notificación, que mantendrá el servicio en segundo plano.

El cálculo de calorías se hace a través de los pasos:

- Se calcula para cada usuario su longitud de paso.
- Se calculan las calorías a partir del número de pasos, la longitud de estos y el peso.

El cálculo de la longitud de paso se hace a través de la edad, la altura, el peso y el sexo del usuario:

```
public void calcularLongitudPaso(){
    double longitud_paso;
    if (sex)
        longitud_paso = height* STEP_MALE;
    else
        longitud_paso = height * STEP_FEMALE;
    longitud_paso *=100; //Convertimos a centímetros
    int edad = getEdad();
    if (edad<EDAD1){
        longitud_paso+=2;
    }else if (edad>=EDAD1 && edad<=(EDAD2-1)){
        longitud_paso+=0;
    }else if (edad>=EDAD2 && edad<=(EDAD3-1)){
        longitud_paso-=1;
    }else if (edad>=EDAD3 && edad<=(EDAD4-1)){
        longitud_paso-=2;
    }else if (edad>=EDAD4){
        longitud_paso-=3;
    }
    if (weight>70){
        longitud_paso+=1;
    }else if (weight<55){
        longitud_paso-=1;
    }
    this.longitud_paso= longitud_paso/100;
}
```

Ilustración 132. Toma de pasos y calorías: Calcular longitud de paso.

```
private static final double STEP_MALE = 0.415;
private static final double STEP_FEMALE = 0.413;

private static final int EDAD1 = 20;
private static final int EDAD2 = 40;
private static final int EDAD3 = 50;
private static final int EDAD4 = 60;
```

Ilustración 133. Toma de pasos y calorías: Constantes para pasos.

Una vez obtenido este valor para cada usuario, las calorías se calculan y actualizan con cada nuevo paso detectado.

```
private BroadcastReceiver updateSteps = (context, intent) → {
    int steps = intent.getIntExtra(StepCounterService.STEPS_EXTRA, defaultValue: 0);

    View customLayout = findViewById(R.id.steps_main);
    if (customLayout != null){
        UserInformationDTO dto = MainWindowFragment.dto;
        if (dto != null){
            calories = (int) (dto.getWeight() * 0.0005 * dto.getLongitud_paso() * steps);
            View customLayout2 = findViewById(R.id.calorie_main);
            TextView st2 = customLayout2.findViewById(R.id.textViewCalories);
            st2.setText(Utils.setStepsString(calories));
        }
        TextView st = customLayout.findViewById(R.id.textViewSteps);
        st.setText(Utils.setStepsString(steps));
    }
};
```

Ilustración 134. Toma de pasos y calorías: updateSteps.

Cuando el servicio envía un paso, las calorías se actualizan a partir de ese dato.

6.3. CSI 3: EJECUCIÓN DE LAS PRUEBAS UNITARIAS

En este apartado se muestran los resultados de las pruebas diseñadas en [5.5.1. Pruebas unitarias](#), que evalúan el correcto funcionamiento de la lógica del sistema. Para cada prueba, se indicará si se ha obtenido el resultado esperado u otro. En el segundo caso se señalará el fallo y la manera en la que se arregló.

6.3.1. Registro de usuario

Registro de usuario	
Prueba	Resultado esperado
Prueba de contraseña.	Las contraseñas que cumplen las condiciones devuelven true, las que no, devuelven false.
	Resultado Obtenido: Correcto.
Prueba de tiene texto.	Los elementos que no tengan texto o contengan texto vacío devolverán false. Los que contengan texto, devolverán true.
	Resultado Obtenido: Correcto.
Prueba de valores introducidos de peso.	Los pesos que no satisfagan las condiciones o que sean incorrectos devolverán false. Los pesos correctos devolverán true.
	Resultado Obtenido: Correcto.



Prueba	Resultado esperado
Prueba de valores introducidos de altura.	Las alturas con formato correcto devolverán true. Las que tengan un formato incorrecto, devolverán false.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Prueba de registrar al usuario.	El sistema registra al usuario, actualizando la base de datos y dejando la sesión iniciada.
	Resultado Obtenido: Correcto.

Tabla 143. Ejecución de las pruebas unitarias: Registro de usuario.

6.3.2. Inicio de sesión

Inicio de sesión	
Prueba	Resultado esperado
Comprobar que la contraseña existe para un usuario concreto.	Codificará la contraseña introducida y la comparará con la que se introduce. Devolverá true si es la misma y false si no son iguales.
	Resultado Obtenido: Correcto.

Tabla 144. Ejecución de las pruebas unitarias: Inicio de sesión.

6.3.3. Actividad principal

Actividad principal	
Prueba	Resultado esperado
Prueba de cerrar sesión.	En la base de datos se almacenan los pasos que realizó el usuario hasta ese momento.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Prueba de registro de la última sesión del usuario.	La última conexión del usuario se actualizará en la base de datos.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Prueba de registro de la última sesión del usuario cuando se inicia la aplicación un día más tarde.	Prueba de registro de la última sesión del usuario.
	Resultado Obtenido: Incorrecto.
	Los pasos no se guardaban de manera correcta porque se actualizaba el registro fuera del método asíncrono. Al reubicar <code>saveRegistry()</code> el problema se solucionó.

Tabla 145. Ejecución de las pruebas unitarias: Actividad principal.

6.3.4. Grabación con los sensores del dispositivo

Grabación con los sensores del dispositivo	
Prueba	Resultado esperado



Prueba de añadir el kilómetro restante.	El elemento que devuelve el método contiene los últimos metros y el tiempo que se tardaron en realizar. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Prueba de añadir el kilómetro restante cuando la actividad dura más de un kilómetro.	El elemento que devuelve el método contiene los últimos metros y el tiempo que se tardaron en realizar. Resultado Obtenido: Correcto.

Tabla 146. Ejecución de las pruebas unitarias: Grabación con los sensores del dispositivo.

6.3.5. Grabación manual

Grabación manual	
Prueba	Resultado esperado
Prueba de guardado de actividad.	La actividad se guarda en la base de datos del usuario y se calculan el ritmo y la velocidad. Resultado Obtenido: Correcto.

Tabla 147. Ejecución de las pruebas unitarias: Grabación manual.

6.3.6. Fragmento con información del usuario

Fragmento con información del usuario	
Prueba	Resultado esperado
Prueba de conseguir información del usuario	Se obtienen los datos del usuario y sus campos están completos y son correctos. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Conseguir actividades de un usuario sin actividades.	El sistema devuelve una lista sin elementos. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Conseguir actividades de un usuario con actividades.	Se devuelve una lista con varias actividades. Resultado Obtenido: Correcto.

Tabla 148. Ejecución de las pruebas unitarias: Fragmento con información del usuario.

6.3.7. Fragmento del historial

Fragmento del historial	
Prueba	Resultado esperado
Prueba de desactivar días sin registros pasados.	El método devuelve una lista vacía. Lo que indica que se desactivaran todos los días del mes en el calendario. Resultado Obtenido: Correcto.
Prueba	Resultado esperado



Prueba de desactivar días con registros pasados.	El método devuelve una lista con los días de los que se guarda registro y que no se van a desactivar en el calendario.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Desactivar días teniendo en cuenta el actual.	El método devuelve una lista con los días de los que se guarda registro, esta lista no contiene el día actual.
	Resultado Obtenido: Correcto.

Tabla 149. Ejecución de las pruebas unitarias: Fragmento del historial.

6.3.8. Fragmento de predicciones

Fragmento de predicciones	
Prueba	Resultado esperado
Conseguir predicciones sin 3 actividades de mínimo 3 kilómetros.	El método devolverá false.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Conseguir predicciones	El método tras hacer una comprobación en la base de datos devolverá true.
	Resultado Obtenido: Correcto.

Tabla 150. Ejecución de las pruebas unitarias: Fragmento de predicciones.

6.3.9. Detalles de una actividad automática

Fragmento del historial	
Prueba	Resultado esperado
Obtención de datos de una actividad de un único punto geográfico.	Se obtendrá un objeto con todos los datos correctos de la actividad.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Obtención de datos de una actividad con muchos puntos geográficos.	Se obtendrá un objeto con todos los datos correctos de la actividad.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Obtención de datos de una actividad con paradas.	Se obtendrá un objeto con todos los datos correctos de la actividad.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Separación si es necesaria. Este método comprueba que una actividad con paradas es tratada correctamente para dividirla en las secciones de puntos geográficos.	De la lista de puntos con pausas, se obtendrán varias listas de puntos.
	Resultado Obtenido: Correcto.

Tabla 151. Ejecución de las pruebas unitarias: Fragmento del historial.

6.3.10. Detalles de una actividad manual

Detalles de una actividad manual	
Prueba	Resultado esperado
Obtención de datos de una actividad manual.	Se obtendrá un objeto con todos los datos correctos de la actividad.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Obtención de datos de una actividad de una hora específica.	Se obtendrá un objeto con todos los datos correctos de la actividad.
	Resultado Obtenido: Correcto.

Tabla 152. Ejecución de las pruebas unitarias: Detalles de una actividad manual.

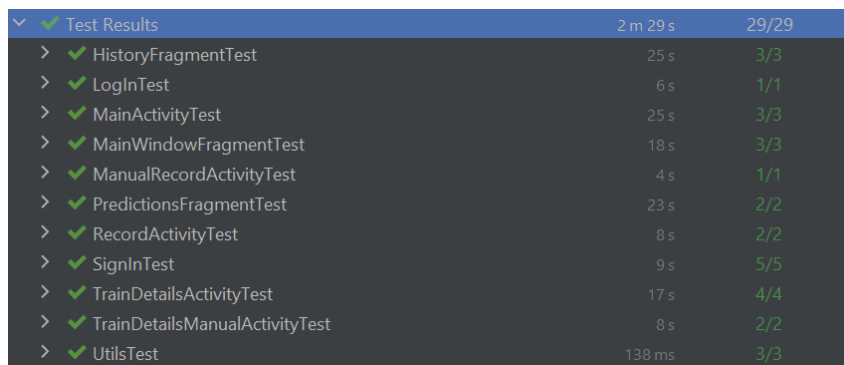
6.3.11. Métodos utils

Métodos utils	
Prueba	Resultado esperado
Conversión de formato de segundos.	Se obtendrá una cadena con el formato esperado (similar al de un cronómetro).
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Conversión de formato de segundos a min/km.	Se obtendrá una cadena con el formato esperado.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Conversión de velocidad a min/km.	Se obtendrá una cadena con el formato esperado.
	Resultado Obtenido: Incorrecto.
	La cadena no tiene el formato esperado. Se modifica el método convertToMinKm para que se obtenga el formato.

Tabla 153. Ejecución de las pruebas unitarias: Métodos utils.

6.3.12. Resultado de las ejecuciones

A continuación, se muestran los resultados de la ejecución de las pruebas en Android Studio.



Test Results	2 m 29 s	29/29
> ✓ HistoryFragmentTest	25 s	3/3
> ✓ LogInTest	6 s	1/1
> ✓ MainActivityTest	25 s	3/3
> ✓ MainWindowFragmentTest	18 s	3/3
> ✓ ManualRecordActivityTest	4 s	1/1
> ✓ PredictionsFragmentTest	23 s	2/2
> ✓ RecordActivityTest	8 s	2/2
> ✓ SignInTest	9 s	5/5
> ✓ TrainDetailsActivityTest	17 s	4/4
> ✓ TrainDetailsManualActivityTest	8 s	2/2
> ✓ UtilsTest	138 ms	3/3

Ilustración 135. Ejecución de las pruebas unitarias: Resultados de las ejecuciones de las pruebas.

Tests	Duration	Xiaomi MI 9
✓ Test Results	2 m 29 s	29/29
HistoryFragmentTest	25 s	3/3
✓ desActivateDatesTest	6 s	✓
✓ desActivateDatesTestWithoutDays	11 s	✓
✓ testDesActivateDatesToday	6 s	✓
LogInTest	6 s	1/1
✓ checkPassword	6 s	✓
MainActivityTest	25 s	3/3
✓ testSaveRegistry	2 s	✓
✓ testLogOut	6 s	✓
✓ testSaveStepsAndRegistry	16 s	✓
MainWindowFragmentTest	18 s	3/3
✓ getInfOfTheUser	6 s	✓
✓ getActivitiesOfTheUserWithActivities	6 s	✓
✓ getActivitiesOfTheUser	6 s	✓
ManualRecordActivityTest	4 s	1/1
✓ testSaveActivity	4 s	✓
PredictionsFragmentTest	23 s	2/2
✓ getPredictionsWithoutThreeActivities	1 s	✓
✓ getPredictions	22 s	✓

Ilustración 136. Ejecución de las pruebas unitarias: Resultado de las ejecuciones de las pruebas 1.

RecordActivityTest	8 s	2/2
✓ testAddTheRestantKilometerLessThanOneKilometer	1 s	✓
✓ testAddTheRestantKilometerMoreThanOneKilometer	6 s	✓
SignInTest	9 s	5/5
✓ registerUser	6 s	✓
✓ checkHeightTest	478 ms	✓
✓ checkWeightTest	588 ms	✓
✓ correctPasswordTest	1 s	✓
✓ hasTextTest	1 s	✓
TrainDetailsActivityTest	17 s	4/4
✓ testDataWithManyPoints	2 s	✓
✓ testData	7 s	✓
✓ testSplitIfIsNecessary	2 s	✓
✓ testDataWithStops	6 s	✓
TrainDetailsManualActivityTest	8 s	2/2
✓ testManualData	2 s	✓
✓ testManualDataWithString	6 s	✓
UtilsTest	138 ms	3/3
✓ stringTimePartialsTest	84 ms	✓
✓ stringTimeTest	34 ms	✓
✓ convertToMinKmTest	20 ms	✓

Ilustración 137. Ejecución de las pruebas unitarias: Resultado de las ejecuciones de las pruebas 2.

Es importante destacar que estas pruebas se realizaron desde el dispositivo especificado en el apartado 5.5 Especificación técnica del plan de pruebas. Y que, para el correcto funcionamiento de las pruebas, se han diseñado con un tiempo de espera para la carga de los datos que depende del dispositivo y la red de conexión a internet, por tanto, si se modifican esas variables los resultados pueden variar.

6.3.13. Problemas y errores

Además de los problemas mencionados en las tablas de los apartados anteriores, han surgido una serie de problemas que han ralentizado el desarrollo de las pruebas unitarias.

El mayor problema fue lograr comprobar el funcionamiento de un activity. Esto fue debido de la insuficiente experiencia con pruebas para dispositivos móviles. Tras un tiempo de investigación y búsqueda se encontró una posible solución.

Finalmente se optó por usar FragmentScenario, que simula el funcionamiento de un fragmento sin tener una actividad que lo contenga. El problema vino de que la aplicación usa los ficheros SharedPreferences para guardar el nombre de usuario y los pasos realizados. Ese fichero solo puede instanciarse desde un contexto específico, por lo que hubo que crear una actividad para rellenar los campos y una vez hecho esto instanciar los fragmentos.

Hasta que se encontró el problema, se recibía un constante NullPointerException por tratar de acceder a los datos de un usuario inexistente (ya que el fichero SharedPreferences no tenía guardado su id).

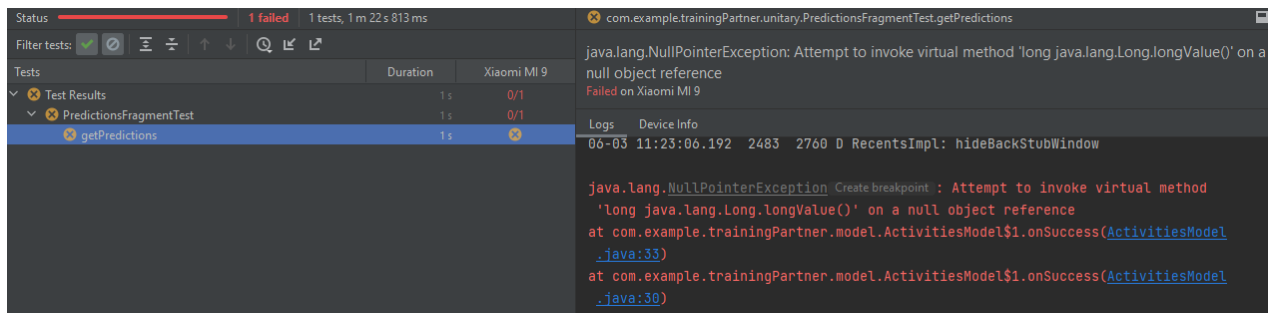


Ilustración 138. Problema 1.

Para solventar este ejemplo se tuvo que realizar lo siguiente:

```
@Before
public void setUp() throws Exception {
    act = myCustom.getActivity();
    Utils.deleteSharedPreferences(act);
    SharedPreferences userPref = act.getSharedPreferences( name: "users", Context.MODE_PRIVATE);
    SharedPreferences.Editor edit = userPref.edit();
    edit.putString("username", "usuarioNoExistente");
    edit.apply();
    UserInformationDTO user = new UserInformationDTO( name: "Pedro", username: "usuarioNoExistente", surname: "Pruebas", new
    fb.collection( collectionPath: "usuarios").document( documentPath: "usuarioNoExistente").set(user.getInformation());

    fragmentScenario= FragmentScenario.launchInContainer(PredictionsFragment.class);
}
```

Ilustración 139. Solución al problema 1.

Se instancia un activity, se introducen los datos en SharedPreferences y posteriormente, se crea el fragmento para realizar las pruebas.

6.4. CSI 4: EJECUCIÓN DE LAS PRUEBAS DE INTEGRACIÓN

Las pruebas de integración prueban el correcto funcionamiento colectivo de los elementos de la aplicación. Se evaluarán las pruebas diseñadas en [5.5.2. Pruebas de integración](#). Los resultados se evaluarán de la misma manera que en el apartado anterior.

En el caso de las pruebas adicionales, se comprobarán si los resultados satisfacen los umbrales de error permitidos y las medidas correctoras en el caso de que lo hagan.

6.4.1. Registro de usuario

Registro de usuario	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Tiene texto. Comprueba que se muestre un mensaje de error si se intenta registrar un usuario sin haber introducido texto en todos los campos.	Se mostrará un mensaje de error específico si se registra un usuario sin haber rellenado todos los campos. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de peso.	Se mostrará un mensaje de error cuando el formato sea incorrecto, y no se mostrará cuando sea válido. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de altura.	Se mostrará un mensaje de error cuando el formato sea incorrecto, y no se mostrará cuando sea válido.

	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de usuario identificado. Comprueba que se notifica al usuario si intenta registrarse con un nombre de usuario que ya está guardado en el sistema.	Se mostrará un mensaje de error específico y el campo de nombre de usuario quedará vacío. Cuando el formato sea correcto se registrará al usuario.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de contraseña.	Se mostrará un mensaje de error específico y el campo de contraseña quedará vacío. Cuando el formato sea correcto se registrará al usuario.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Mover a inicio de sesión.	Si el usuario pulsa el botón se le dirigirá a la pantalla de inicio de sesión. Al volver a la de registro, todos los campos estarán vacíos.
	Resultado Obtenido: Correcto.

Tabla 154. Ejecución de las pruebas de integración: Registro de usuario.

6.4.2. Inicio de sesión

Inicio de sesión	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Tiene texto. Comprueba que se muestre un mensaje de error si se intenta iniciar sesión sin haber introducido texto en todos los campos.	Se mostrará un mensaje de error específico si se inicia sesión sin haber rellenado todos los campos.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de usuario existente.	Se mostrará un mensaje de error. En el momento que se introduzca un nombre correcto, se iniciará la sesión.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de altura.	Se mostrará un mensaje de error cuando el formato sea incorrecto, y no se mostrará cuando sea válido.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de contraseña.	Se mostrará un mensaje de error. En el momento en el que se introduzca la contraseña correcta, se iniciará la sesión.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comprobación de contraseña.	Se mostrará un mensaje de error específico y el campo de contraseña quedará vacío. Cuando el formato sea correcto se registrará al usuario.

	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Mover a registro.	Al pulsar el botón se redirigirá al usuario a la pantalla de registro, si se vuelve a la pantalla de inicio de sesión, todos los campos estarán vacíos.
	Resultado Obtenido: Correcto.

Tabla 155. Ejecución de las pruebas de integración: Inicio de sesión.

6.4.3. Actividad principal

Actividad principal	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están. Los datos del usuario aparecen correctamente tras el tiempo de carga.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Cambiar de fragmento. Comprueba el funcionamiento del menú inferior que cambia el fragmento que se muestra en la actividad principal.	Al pulsar los botones del menú inferior se cambiará al fragmento esperado. Si se pulsa el botón atrás se volverá al fragmento anterior.
	Resultado Obtenido: Incorrecto.
	Al pulsar el botón de atrás se recargaba el mismo fragmento y no se cambiaba. Se solucionó guardando registro del anterior fragmento seleccionado.

Tabla 156. Ejecución de las pruebas de integración: Actividad principal.

6.4.4. Grabar actividades con los sensores del dispositivo

RecordActivity	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Comenzar actividad.	La actividad se inicia y se pausa según lo pulsado en los botones de la pantalla. Los botones se mostrarán o no dependiendo de si la actividad está iniciada o pausada.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Pulsar hacia atrás con la actividad no iniciada.	Se muestra la pantalla principal y no se ha guardado ninguna actividad.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Pulsar hacia atrás con la actividad iniciada.	Se muestra la pantalla principal y no se ha guardado ninguna actividad.
	Resultado Obtenido: Correcto.
Prueba	Resultado esperado



Pulsar hacia atrás con la actividad pausada.	Se muestra la pantalla principal y no se ha guardado ninguna actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Guardar actividad. Comprueba el funcionamiento de la finalización de la actividad.	Se para y guarda la actividad y se muestran los detalles de esta. Resultado Obtenido: Incorrecto. Se iniciaba TrainDetailsActivity demasiado pronto. Para asegurar que se mostrase la actividad una vez ha sido guardada en la base de datos, se añadió un callback que cambia de activity cuando la actividad había sido guardada.

Tabla 157. Ejecución de las pruebas de integración: Grabar actividades con los sensores del dispositivo.

6.4.5. Grabar actividad manualmente

ManualRecordActivity	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Pulsar hacia atrás.	Se redirigirá al usuario a la pantalla principal y no se habrá guardado ninguna actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Activar botón. Comprueba si se activa correctamente el botón de guardar al interactuar con los elementos de la pantalla.	El botón de guardar actividad se activará cuando los valores de distancia y tiempo sean distintos de 0. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
No activar botón. Comprueba que, aunque se de valor al tiempo, si la distancia es 0, el botón no se activa.	El botón de guardar permanecerá desactivado. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Guardar actividad.	Se guarda la actividad y se dirige al usuario a los detalles de esta. Resultado Obtenido: Correcto.

Tabla 158. Ejecución de las pruebas de integración: Grabar actividad manualmente.

6.4.6. Fragmento con información del usuario

MainWindowFragment

Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Cambiar peso. Comprueba el funcionamiento de cambiar el peso del usuario.	El peso se actualizará en la interfaz gráfica y en la base de datos. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Cambiar altura. Comprueba el funcionamiento de cambiar la altura del usuario.	La altura se actualizará en la interfaz gráfica y en la base de datos. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Mostrar actividad. Comprueba que al pulsar sobre un elemento actividad del fragmento principal se muestren los detalles de esta.	Se muestran los detalles de la actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Cerrar sesión. Comprueba el funcionamiento de cerrar la sesión.	Se cierra la sesión, se actualiza el registro en la base de datos. Resultado Obtenido: Correcto.

Tabla 159. Ejecución de las pruebas de integración: Fragmento con información del usuario.

6.4.7. Fragmento del historial

HistoryFragment	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están. Sólo están activados los días de los que hay registro. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Cambiar mes. Testea el funcionamiento de del cambio del mes del calendario.	El mes se cambia correctamente. Cambian los días activados. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Ver un día histórico. Comprueba el funcionamiento de ver un día histórico.	Se muestran los detalles del día seleccionado. Resultado Obtenido: Correcto.

Tabla 160. Ejecución de las pruebas de integración: Fragmento del historial.

6.4.8. Fragmento de un día histórico

HistoricalDayFragment	
Prueba	Resultado esperado
Todo en su lugar.	Los elementos que deben de estar visibles lo están.

	Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Abrir actividad. Comprueba el funcionamiento de ver los detalles de una actividad que se realizó en un día pasado.	Se muestran los detalles de esa actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Pulsar hacia atrás. Comprueba el funcionamiento de pulsar el botón de ir hacia atrás.	Se redirige al usuario a la pantalla de visión del historial. Resultado Obtenido: Correcto.

Tabla 161. Ejecución de las pruebas de integración: Fragmento de un día histórico.

6.4.9. Fragmento de predicciones

PredictionsFragment	
Prueba	Resultado esperado
Todo en su lugar cuando no hay suficientes actividades para realizar predicciones.	Se muestran todos los elementos, entre ellos el mensaje explicativo de que no se pueden realizar aún predicciones. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Todo en su lugar cuando hay suficientes actividades para realizar predicciones.	Se muestran todos los elementos, entre ellos las predicciones. Resultado Obtenido: Correcto.

Tabla 162. Ejecución de las pruebas de integración: Fragmento de predicciones.

6.4.10. Detalles de una actividad automática

Train Details	
Prueba	Resultado esperado
Detalles de una actividad de un solo punto.	Se muestran todos los elementos correctamente. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Detalles de una actividad de varios puntos geográficos.	Se muestran todos los elementos correctamente. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Detalles de una actividad de varios puntos geográficos y con paradas.	Se muestran todos los elementos correctamente. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Eliminar una actividad. Comprueba el proceso de eliminación de una actividad.	La actividad se elimina y se redirige al usuario a la pantalla principal, donde no se muestra la actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Guardar actividad. Comprueba el proceso de guardar la actividad.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad. Resultado Obtenido: Correcto.

Prueba	Resultado esperado
Pulsar el botón hacia atrás. Comprueba el proceso que se ejecuta tras pulsar el botón de atrás.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad. Resultado Obtenido: Correcto.

Tabla 163. Ejecución de las pruebas de integración: Detalles de una actividad automática.

6.4.11. Detalles de una actividad manual

Train Details Manual Activity	
Prueba	Resultado esperado
Todo en su lugar.	Todos los elementos se muestran correctamente. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Eliminar una actividad. Comprueba el proceso de eliminación de una actividad.	La actividad se elimina y se redirige al usuario a la pantalla principal, donde no se muestra la actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Guardar actividad. Comprueba el proceso de guardar la actividad.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad. Resultado Obtenido: Correcto.
Prueba	Resultado esperado
Pulsar el botón hacia atrás. Comprueba el proceso que se ejecuta tras pulsar el botón de atrás.	Se redirige al usuario a la pantalla principal, donde se muestra la actividad. Resultado Obtenido: Correcto.

Tabla 164. Ejecución de las pruebas de integración: Detalles de una actividad manual.

6.4.12. Pruebas adicionales

A continuación, se muestran los resultados de las pruebas adicionales.

6.4.12.1. Prueba de pasos

Para esta prueba se han realizado varios tests de 300 pasos. Es necesario destacar que la toma de pasos se hace mediante el sensor acelerómetro, para asegurar esta funcionalidad a todo tipo de dispositivo Android.

El mayor problema de este método es su inexactitud, un zarandeo con el dispositivo en la mano podría ser interpretado como varios pasos. En este sistema se ha priorizado que la toma con el dispositivo en la mano sea lo más exacta posible, siempre aceptando las limitaciones del método. Consideraremos un resultado positivo aquel que fluctúe en 20 pasos del real.

Se han realizado 300 pasos exactamente con el dispositivo en el bolsillo y este ha sido el resultado:

PASOS REALES	CONTEO	RESULTADO
300	 <p>  PASOS: 317  CALORIAS: 8 </p>	POSITIVO

Tabla 165. Prueba de pasos 1.

Como se puede comprobar el resultado es muy prometedor, pero para asegurar los resultados se hacen otras dos pruebas de 300 pasos:







PASOS REALES	CONTEO	RESULTADO
300	 <p>  PASOS: 290  CALORIAS: 7 </p>	POSITIVO
300	 <p>  PASOS: 306  CALORIAS: 8 </p>	POSITIVO

Tabla 166. Prueba de pasos 2.

Los resultados indican que el conteo de pasos se realiza correctamente cuando el dispositivo se encuentra en el bolsillo, ya que estas pequeñas variaciones se presentan hasta en los medidores de muñeca de otros sistemas de mayor presupuesto y recursos.

Como contra, el conteo de pasos con el dispositivo en la mano es muy impreciso, se han priorizado valores coherentes en parado y con el móvil en el bolsillo a costa de conseguir valores mucho menores en el conteo de pasos con el móvil en la mano. A continuación, se muestra un conteo de 100 pasos sujetando el dispositivo con el brazo extendido a la altura de la cadera.

PASOS REALES	CONTEO	RESULTADO
100		NEGATIVO

Tabla 167. Prueba de pasos 3.

Como se pueden comprobar, para estos casos los resultados son mucho peores, pero como se ha comentado anteriormente, se ha decidido priorizar la coherencia del conteo con el dispositivo en el bolsillo. Si se deseara mejorar el cálculo en el caso de que el móvil sea sujetado con el brazo, sólo sería necesario reducir el umbral de movimiento. Sin embargo, habría que tener en cuenta que el número de pasos percibidos cuando el dispositivo está en el bolsillo aumentaría.

6.4.12.2. Prueba de grabado de la actividad

A continuación, se muestran las pruebas de grabado de actividad, para ello se van a comparar 3 actividades grabadas con la aplicación y con el dispositivo Garmin Forerunner 245.

6.4.12.2.1. Actividad 1

6.4.12.2.1.1. Mapa

TRAINING PARTNER	GARMIN CONNECT

Tabla 168. Mapa actividad 1.

Como se puede comprobar el mapa creado es prácticamente igual en los dos casos. Aunque es cierto que la cosa cambia si se hace zoom, debido a que Training Partner guarda muchos menos puntos geográficos que Garmin.

6.4.12.2.1.2. Datos generales

TRAINING PARTNER	GARMIN CONNECT
<p>DISTANCIA 5.29 km</p> <p>VELOCIDAD MEDIA 11.41 km/h</p> <p>TIEMPO 00:27:49</p> <p>RITMO MEDIO 05:15 min/km</p>	<p>5,01 Distancia (km)</p> <p>27:38 Tiempo</p> <p>293 Calorías</p> <p>5:31 Ritmo (min/km)</p> <p>154 Ascenso total (m)</p>

Tabla 169. Datos generales actividad 1.

Las mediciones varían un poco, pero se mantienen dentro de los valores aceptables. Es cierto que el tiempo es un poco mayor en el caso de Training Partner, pero es debido a que se tarda algo de tiempo en guardar y extraer el dispositivo del bolsillo del pantalón deportivo.

En cuanto a la leve diferencia de distancias y ritmos, puede ser debida al hecho de que la aplicación creada en este proyecto toma un menor número de puntos geográficos.

6.4.12.2.1.3. Parciales

TRAINING PARTNER	GARMIN CONNECT																																																								
<p>PARCIALES</p> <table border="1"> <thead> <tr> <th>Vuelta</th> <th>Tiempo</th> <th>Distancia</th> <th>Ritmo</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>00:06:00</td> <td>1 km</td> <td>06:00 min/km</td> </tr> <tr> <td>2</td> <td>00:04:55</td> <td>1 km</td> <td>04:55 min/km</td> </tr> <tr> <td>3</td> <td>00:05:05</td> <td>1 km</td> <td>05:05 min/km</td> </tr> <tr> <td>4</td> <td>00:04:50</td> <td>1 km</td> <td>04:50 min/km</td> </tr> <tr> <td>5</td> <td>00:05:07</td> <td>1 km</td> <td>05:07 min/km</td> </tr> <tr> <td>6</td> <td>00:01:52</td> <td>0.29 m</td> <td>06:25 min/km</td> </tr> </tbody> </table>	Vuelta	Tiempo	Distancia	Ritmo	1	00:06:00	1 km	06:00 min/km	2	00:04:55	1 km	04:55 min/km	3	00:05:05	1 km	05:05 min/km	4	00:04:50	1 km	04:50 min/km	5	00:05:07	1 km	05:07 min/km	6	00:01:52	0.29 m	06:25 min/km	<table border="1"> <thead> <tr> <th>Vuelta</th> <th>Tiempo</th> <th>Distancia km</th> <th>Ritmo medio min/km</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>5:48,4</td> <td>1,00</td> <td>5:48</td> </tr> <tr> <td>2</td> <td>5:34,3</td> <td>1,00</td> <td>5:34</td> </tr> <tr> <td>3</td> <td>5:22,5</td> <td>1,00</td> <td>5:22</td> </tr> <tr> <td>4</td> <td>5:31,0</td> <td>1,00</td> <td>5:31</td> </tr> <tr> <td>5</td> <td>5:19,6</td> <td>1,00</td> <td>5:20</td> </tr> <tr> <td>6</td> <td>0:02,5</td> <td>0,01</td> <td>4:40</td> </tr> </tbody> </table>	Vuelta	Tiempo	Distancia km	Ritmo medio min/km	1	5:48,4	1,00	5:48	2	5:34,3	1,00	5:34	3	5:22,5	1,00	5:22	4	5:31,0	1,00	5:31	5	5:19,6	1,00	5:20	6	0:02,5	0,01	4:40
Vuelta	Tiempo	Distancia	Ritmo																																																						
1	00:06:00	1 km	06:00 min/km																																																						
2	00:04:55	1 km	04:55 min/km																																																						
3	00:05:05	1 km	05:05 min/km																																																						
4	00:04:50	1 km	04:50 min/km																																																						
5	00:05:07	1 km	05:07 min/km																																																						
6	00:01:52	0.29 m	06:25 min/km																																																						
Vuelta	Tiempo	Distancia km	Ritmo medio min/km																																																						
1	5:48,4	1,00	5:48																																																						
2	5:34,3	1,00	5:34																																																						
3	5:22,5	1,00	5:22																																																						
4	5:31,0	1,00	5:31																																																						
5	5:19,6	1,00	5:20																																																						
6	0:02,5	0,01	4:40																																																						

Tabla 170. Parciales actividad 1.

Es posible que en este apartado es donde más diferencias se encuentren, debido a las limitaciones de localización que tiene la aplicación. Aun así, los parciales no tienen grandes diferencias y se mantienen dentro de valores coherentes. Hay que destacar que suele haber diferencias entre los parciales percibidos por diferentes dispositivos Garmin.

6.4.12.2.1.4. Conclusión

Con las comparaciones anteriores se puede concluir que el resultado de la prueba de esta actividad es **SATISFACTORIO**.

6.4.12.2.2. Actividad 2

En esta ocasión se compararán dos actividades de menor distancia.

6.4.12.2.2.4. Mapa



Tabla 171. Mapa actividad 2.

Al igual que en la actividad 1, los mapas tomados con ambas aplicaciones son muy similares. Al ser este un entrenamiento más corto, es más visible que hay puntos que se cogen con menor precisión.

6.4.12.2.2.5. Datos generales

TRAINING PARTNER	GARMIN CONNECT								
<table border="0"> <tr> <td>DISTANCIA</td> <td>VELOCIDAD MEDIA</td> </tr> <tr> <td>2.42 km</td> <td>3.13 km/h</td> </tr> <tr> <td>TIEMPO</td> <td>RITMO MEDIO</td> </tr> <tr> <td>00:46:19</td> <td>19:09 min/km</td> </tr> </table>	DISTANCIA	VELOCIDAD MEDIA	2.42 km	3.13 km/h	TIEMPO	RITMO MEDIO	00:46:19	19:09 min/km	
DISTANCIA	VELOCIDAD MEDIA								
2.42 km	3.13 km/h								
TIEMPO	RITMO MEDIO								
00:46:19	19:09 min/km								

Tabla 172. Datos generales actividad 2.

Los datos generales vuelven a ser muy parecidos. Es cierto que en esta ocasión al ser una distancia más corta la velocidad y el ritmo varían más, pero están dentro de los valores racionales.

6.4.12.2.2.6. Parciales

TRAINING PARTNER				GARMIN CONNECT			
Vuelta	Tiempo	Distancia	Ritmo	Vuelta	Tiempo	Distancia km	Velocidad media km/h
1	00:18:57	1 km	18:57 min/km	1	21:50,7	1,00	2,7
2	00:21:00	1 km	21:00 min/km	2	22:09,2	1,00	2,7
3	00:06:22	0.42 m	15:09 min/km	3	2:21,4	0,14	3,5

Tabla 173. Parciales actividad 2.

Al igual que en los datos generales, tenemos valores con una diferencia considerable pero aceptables.

6.4.12.2.2.7. Conclusión

Con las comparaciones realizadas anteriormente, se puede concluir que para actividades cortas el comportamiento de grabación de actividades es **VÁLIDO**.

6.4.12.2.3. Actividad 3

En esta ocasión se compararán dos actividades de una distancia más larga.

6.4.12.2.3.4. Mapa

TRAINING PARTNER	GARMIN CONNECT

Tabla 174. Mapa actividad 3.

En este caso se puede comprobar que los mapas son prácticamente iguales. Incluso se captaron en ambos una pequeña vuelta a mitad de la actividad.

6.4.12.2.3.5. Datos generales


TRAINING PARTNER	GARMIN CONNECT
<p>DISTANCIA 10.55 km</p> <p>VELOCIDAD MEDIA 12.35 km/h</p> <p>TIEMPO 00:51:15</p> <p>RITMO MEDIO 04:51 min/km</p>	

Tabla 175. Datos generales actividad 3.

Los datos generales son más parecidos que nunca. El tiempo tomado con Training Partner es un poco mayor ya que se tarda en guardar y extraer el dispositivo del bolsillo en la ropa deportiva. La distancias, velocidades y ritmos son prácticamente iguales.

6.4.12.2.3.6. Parciales

TRAINING PARTNER				GARMIN CONNECT			
PARCIALES							
Vuelta	Tiempo	Distancia	Ritmo				
1	00:04:53	1 km	04:53 min/km	1	5:10,5	1,00	5:10
2	00:05:23	1 km	05:23 min/km	2	4:44,2	1,00	4:44
3	00:05:12	1 km	05:12 min/km	3	4:53,9	1,00	4:54
4	00:04:38	1 km	04:38 min/km	4	4:49,9	1,00	4:50
5	00:05:27	1 km	05:27 min/km	5	4:46,2	1,00	4:46
6	00:04:34	1 km	04:34 min/km	6	4:43,6	1,00	4:44
7	00:05:14	1 km	05:14 min/km	7	4:40,2	1,00	4:40
8	00:04:47	1 km	04:47 min/km	8	4:49,7	1,00	4:50
9	00:04:43	1 km	04:43 min/km	9	4:36,5	1,00	4:36
10	00:05:14	1 km	05:14 min/km	10	4:35,3	1,00	4:35
11	00:01:10	0.55 m	02:07 min/km	11	2:04,5	0,41	5:01

Tabla 176. Parciales actividad 3.

Los parciales varían un poco, pero todos tienen en ambas aplicaciones valores racionales y similares. Es normal que usando distintas precisiones y sensores GPS, los valores no sean iguales. Estas diferencias se presentan en relojes profesionales de distintas marcas o incluso de la misma.

6.4.12.2.3.7. Conclusión

Con las comparaciones realizadas anteriormente, se puede concluir que para actividades largas el comportamiento es **MUY SATISFACTORIO**.

6.4.12.2.4. Conclusión de las pruebas de grabado

A partir de las pruebas anteriores, se concluye que el comportamiento de la aplicación en lo que refiere al grabado de actividades con los sensores del dispositivo es válido y en la gran mayoría de ocasiones funciona tan correctamente como otros sistemas con más recursos.

6.4.12.3. Prueba de predicciones

A continuación, se detallan los resultados de las pruebas de predicciones. Para cada uno de los usuarios de prueba, se indicarán los tiempos introducidos, así como sus predicciones y su grado de satisfacción.

TIEMPOS INTRODUCIDOS USUARIO 1	
DISTANCIA	RITMO
21.097 km	03:52 min/km
3 km	03:24 min/km
5 km	03:27 min/km
PREDICCIONES PARA EL USUARIO 1	
DISTANCIA	RITMO
1 km	03:15 min/km
10 km	03:34 min/km
Media maratón	03:45 min/km
Maratón	04:12 min/km

Tabla 177. Prueba de predicciones usuario 1.

Grado de satisfacción con las predicciones USUARIO 1. ¿Cómo de factible es realizar esos tiempos?
PROBABLE

Tabla 178. Grado de satisfacción usuario 1.

TIEMPOS INTRODUCIDOS USUARIO 2	
DISTANCIA	RITMO
12.100 km	04:57 min/km
8 km	04:57 min/km
10 km	04:58 min/km
PREDICCIONES PARA EL USUARIO 2	
DISTANCIA	RITMO
1 km	04:42 min/km
10 km	04:42 min/km
Media maratón	05:13 min/km
Maratón	05:50 min/km

Tabla 179. Prueba de predicciones usuario 2.

Grado de satisfacción con las predicciones USUARIO 2.

¿Cómo de factible es realizar esos tiempos?

PROBABLE

Tabla 180. Grado de satisfacción usuario 2.

TIEMPOS INTRODUCIDOS USUARIO 3	
DISTANCIA	RITMO
3 km	04:30 min/km
5 km	04:36 min/km
42,197 km	05:19 min/km
PREDICCIONES PARA EL USUARIO 3	
DISTANCIA	RITMO
1 km	04:16 min/km
10 km	04:46 min/km
Media maratón	05:00 min/km
Maratón	05:37 min/km

Tabla 181. Prueba de predicciones usuario 3.

Grado de satisfacción con las predicciones USUARIO 3.

¿Cómo de factible es realizar esos tiempos?

MUY PROBABLE

Tabla 182. Grado de satisfacción usuario 3.

Los resultados de estas pruebas son suficientemente buenos como para concluir que el sistema de predicciones funciona correctamente y da valores cercanos a los posibles.

6.4.13. Resultados de las ejecuciones

Tests	Duration	Xiaomi MI 9
Test Results	8 m 30 s	48/48
HistoricalDayFragmentIntegrationTest	46 s	3/3
HistoryFragmentIntegrationTest	42 s	3/3
LogInIntegrationTest	31 s	5/5
MainActivityIntegrationTest	34 s	2/2
MainWindowFragmentIntegrationTest	1 m 31 s	5/5
ManualRecordActivityIntegrationTest	18 s	5/5
PredictionsFragmentIntegrationTest	44 s	2/2
RecordActivityIntegrationTest	1 m 3 s	6/6
SignInIntegrationTest	1 m 11 s	7/7
TrainDetailsActivityIntegrationTest	33 s	6/6
TrainDetailsManualActivityIntegrationTest	33 s	4/4

Ilustración 140. Pruebas de integración: Resultado de las ejecuciones.

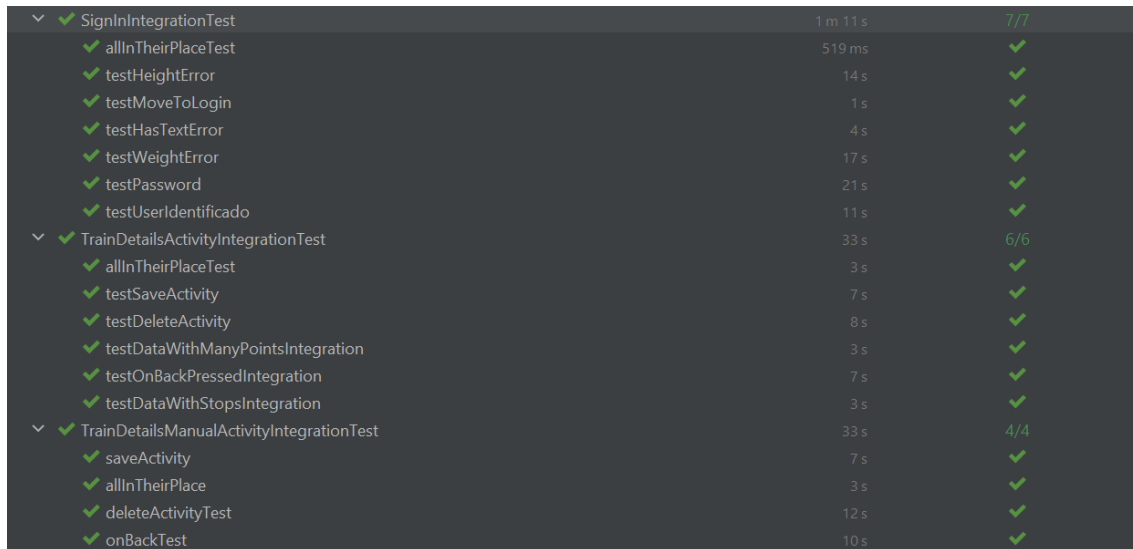


Test Results	8 m 30 s	48/48
HistoricalDayFragmentIntegrationTest	46 s	3/3
allInTheirPlaceTest	15 s	✓
onBackPressedTest	15 s	✓
openActivityTest	15 s	✓
HistoryFragmentIntegrationTest	42 s	3/3
changeMonthTest	14 s	✓
allInTheirPlace	13 s	✓
goToHistoricalDayTest	14 s	✓
LogInIntegrationTest	31 s	5/5
allInTheirPlaceTest	1 s	✓
userNotExists	11 s	✓
testMoveToSignIn	2 s	✓
wrongPasswordTest	11 s	✓
allFieldsCompleteTest	5 s	✓
MainActivityIntegrationTest	34 s	2/2
changeFragment	20 s	✓
allInTheirPlace	13 s	✓

Ilustración 141. Pruebas de integración: Resultado de las pruebas 1.

MainWindowFragmentIntegrationTest	1 m 31 s	5/5
changeWeightTest	20 s	✓
allInTheirPlace	14 s	✓
showActivityTest	15 s	✓
logOutTest	20 s	✓
changeHeightTest	20 s	✓
ManualRecordActivityIntegrationTest	18 s	5/5
allInTheirPlace	896 ms	✓
buttonNotActivateTest	4 s	✓
onBackPressedTest	4 s	✓
buttonActivateTest	4 s	✓
saveManualActivityTest	4 s	✓
PredictionsFragmentIntegrationTest	44 s	2/2
allInTheirPlaceAddActivity	26 s	✓
allInTheirPlace	18 s	✓
RecordActivityIntegrationTest	1 m 3 s	6/6
onBackPressedActivityInitiatedTest	9 s	✓
startActivityTest	9 s	✓
allInTheirPlaceTest	7 s	✓
saveActivityTest	15 s	✓
onBackPressedActivityNotInitiatedTest	10 s	✓
onBackPressedActivityPausedTest	11 s	✓

Ilustración 142. Pruebas de integración: Resultado de las pruebas 2.



Test Name	Duration	Status
SignInIntegrationTest	1 m 11 s	7/7
allInTheirPlaceTest	519 ms	✓
testHeightError	14 s	✓
testMoveToLogin	1 s	✓
testHasTextError	4 s	✓
testWeightError	17 s	✓
testPassword	21 s	✓
testUserIdentificado	11 s	✓
TrainDetailsActivityIntegrationTest	33 s	6/6
allInTheirPlaceTest	3 s	✓
testSaveActivity	7 s	✓
testDeleteActivity	8 s	✓
testDataWithManyPointsIntegration	3 s	✓
testOnBackPressedIntegration	7 s	✓
testDataWithStopsIntegration	3 s	✓
TrainDetailsManualActivityIntegrationTest	33 s	4/4
saveActivity	7 s	✓
allInTheirPlace	3 s	✓
deleteActivityTest	12 s	✓
onBackPressed	10 s	✓

Ilustración 143. Pruebas de integración: Resultado de las pruebas 3.

Al igual que se mencionó en el apartado [6.3.12. Resultado de las ejecuciones](#), estas pruebas han sido realizadas con unos tiempos de espera diseñados para una conexión de internet y un dispositivo específicos. Al variar estas variables, los resultados pueden variar.

6.4.14. Problemas y errores

A parte de los errores mencionados anteriormente, al realizar las pruebas de implementación se encontraron una serie de problemas y errores.

El problema principal fue la dificultad de hacer pruebas sobre un sistema asíncrono, hubo que introducir una serie de pausas para esperar a que los datos se carguen en la aplicación.

También se encontraron problemas parecidos a los de [6.3.13. Problemas y errores](#) ya que hubo que antes de instanciar una actividad hay que instanciar otra para rellenar los datos del fichero SharedPreferences.



```
@Before
public void setUp() throws InterruptedException {
    SplashActivity act = signRule.getActivity();
    SharedPreferences userPref = act.getApplicationContext().getSharedPreferences( name: "users", Context.MODE_PRIVATE);
    SharedPreferences.Editor edit = userPref.edit();
    edit.putString("username", "usuarioNoExistente");
    edit.apply();
    user = new UserInformationDTO( name: "Pedro", username: "usuarioNoExistente", surname: "Pruebas", new Date(),
    fb.collection( collectionPath: "usuarios").document( documentPath: "usuarioNoExistente").set(user.getInformation());
    SharedPreferences stepsPref = act.getApplicationContext().getSharedPreferences( name: "stepCounter", Context.MODE_PRIVATE);
    SharedPreferences.Editor editor = stepsPref.edit();
    editor.putInt("stepCount", 15);
    editor.apply();
    signRule.finishActivity();
    Thread.sleep( millis: 2000);
    mainRule.launchActivity( startIntent: null);
    MainActivity = mainRule.getActivity();
}
```

Ilustración 144. Problemas pruebas de integración.

Finalmente, se encontraron problemas para interactuar con algunos elementos del sistema como el *MaterialCalendarView* o los *Spinners*. Por ejemplo, en el caso de los segundos, en *ManualRecordActivity* cuando se quería probar el funcionamiento de activar el botón en el momento que el spinner cambiaba, se tuvo que falsear un deslizamiento para que se ejecutara el listener del componente. Esta acción de deslizar se ejecutaba en un hilo a parte de la prueba, por lo que se debía tener en cuenta que el test podía terminar antes que la propia acción.

6.5. CSI 5: EJECUCIÓN DE PRUEBAS DEL SISTEMA

6.5.1. Ejecución de las pruebas de usabilidad

A continuación, se muestran los resultados de las pruebas de usabilidad del sistema.

6.5.1.1. Pruebas sobre el usuario

¿Qué edad tienes?

4 respuestas

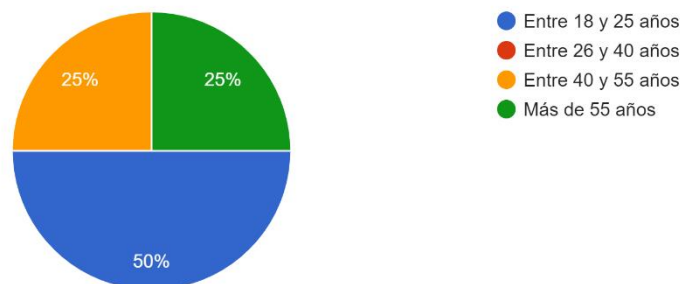


Ilustración 145. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 1.

¿Cuántas horas dedicas diariamente al uso del dispositivo móvil?

4 respuestas

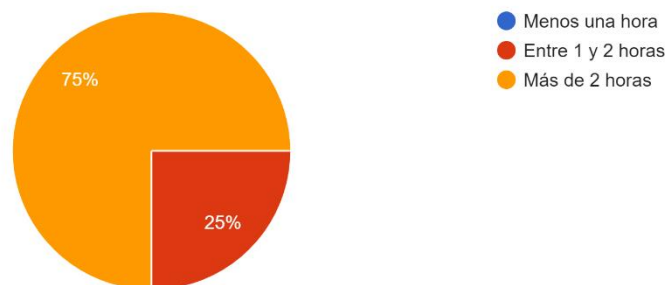


Ilustración 146. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 2.

¿Ha usado anteriormente aplicaciones para el seguimiento de métricas deportivas?

4 respuestas

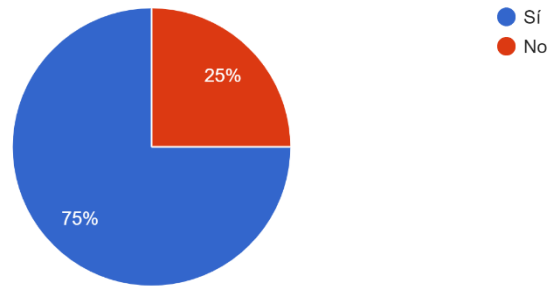


Ilustración 147. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 3.

En caso afirmativo, por favor indique cuales

3 respuestas

- Mi fit
- Garmin connect
- Garmin connect y strava

Ilustración 148. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 4.

¿Qué busca en una aplicación de estas características?

4 respuestas

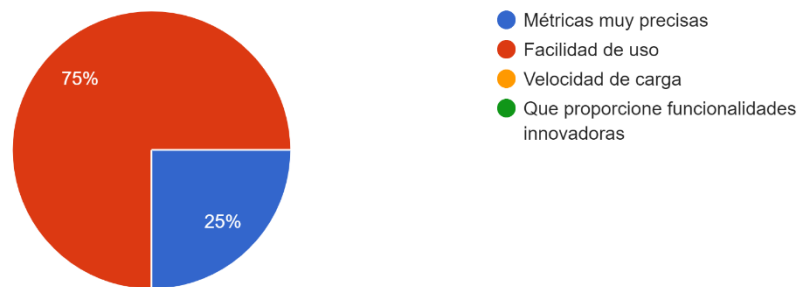


Ilustración 149. Ejecución de las pruebas de usabilidad: Pruebas sobre el usuario 5.

6.5.1.2. Actividades guiadas

Registrarse en el sistema

4 respuestas

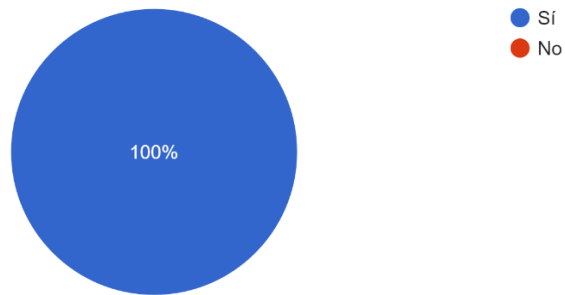


Ilustración 150. Ejecución de las pruebas de usabilidad: Actividades guiadas 1.

Registrarse en el sistema

4 respuestas

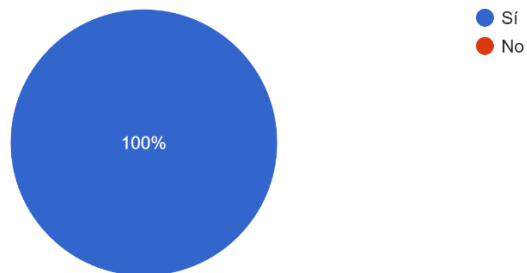
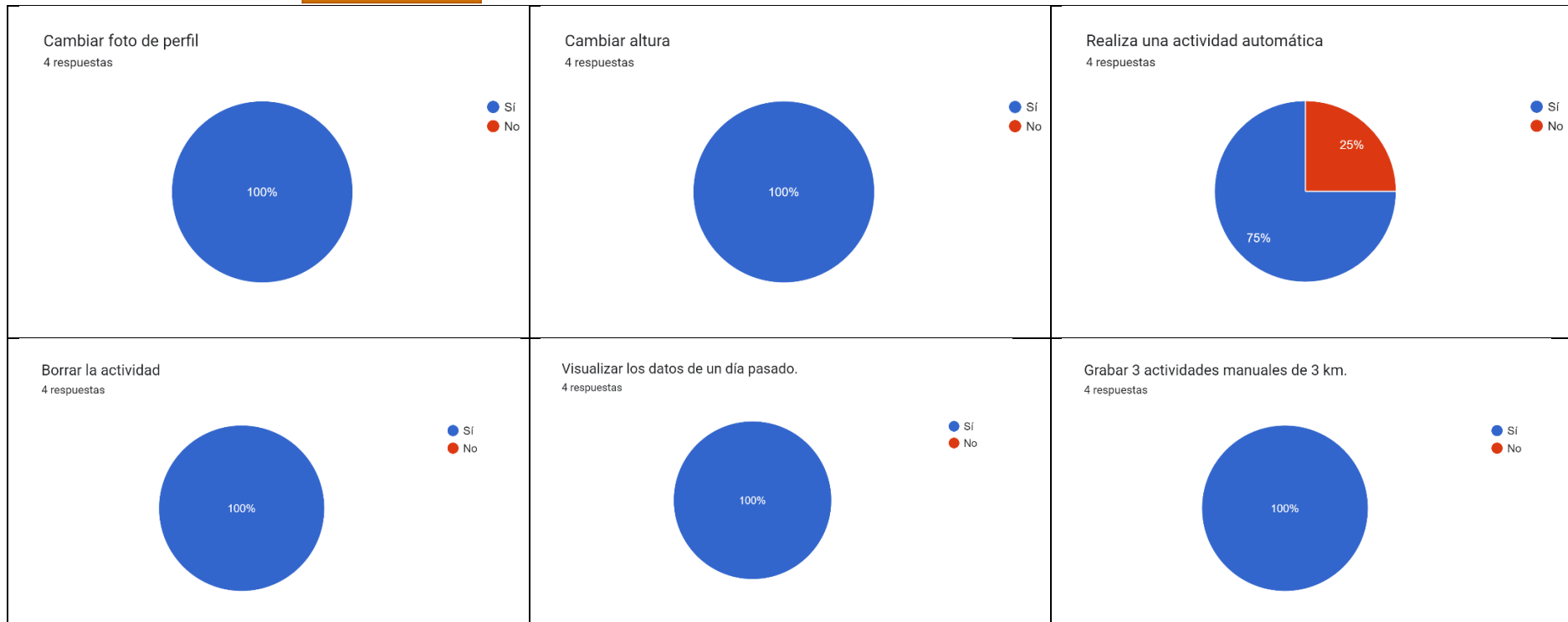


Ilustración 151. Ejecución de las pruebas de usabilidad: Actividades guiadas 2.





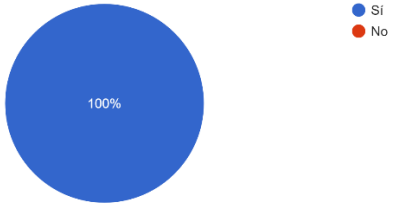
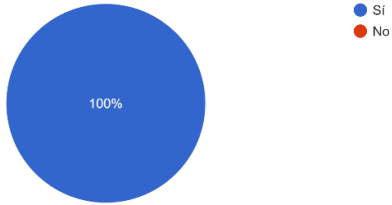
<p>Visualizar las predicciones. 4 respuestas</p>  <p>● Si ● No</p>	<p>Cerrar sesión 4 respuestas</p>  <p>● Si ● No</p>	
---	---	--

Tabla 183. Ejecución de las pruebas de usabilidad: Actividades guiadas.

Problemas encontrados

1 respuesta

Un poco de dificultad al moverse por la interfaz

Ilustración 152. Ejecución de las pruebas de usabilidad: Actividades guiadas 4.

6.5.1.3. Cuestiones de usabilidad

Valora la interfaz gráfica de la aplicación.

4 respuestas

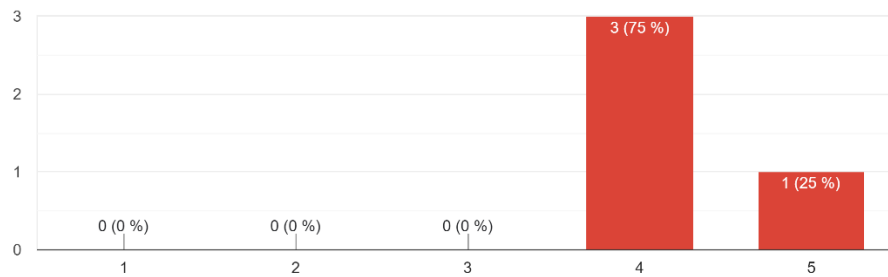


Ilustración 153. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 1.

Valora la usabilidad de la aplicación

4 respuestas

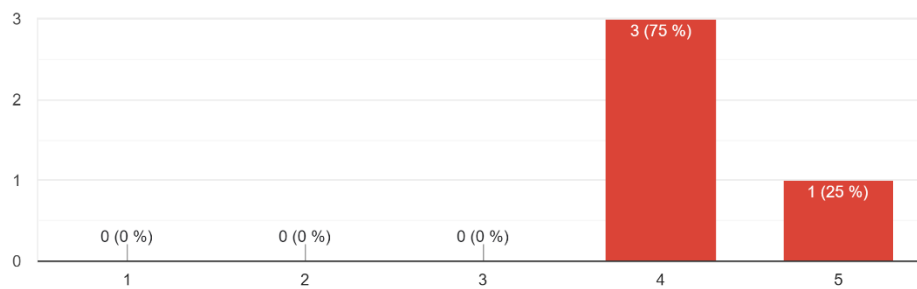


Ilustración 154. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 2.

Valora el funcionamiento de la aplicación

4 respuestas

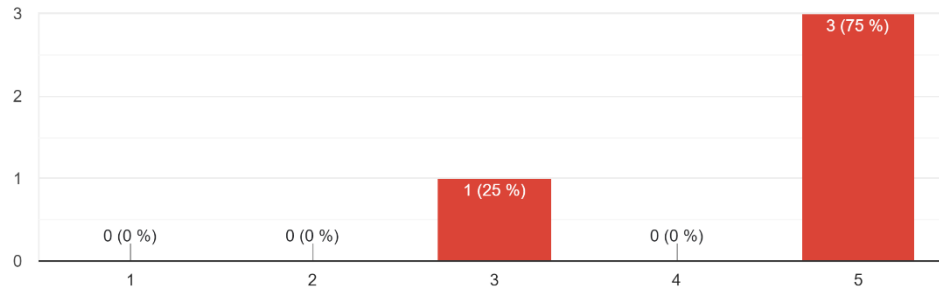


Ilustración 155. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 3.

Valora la navegabilidad de la aplicación

4 respuestas

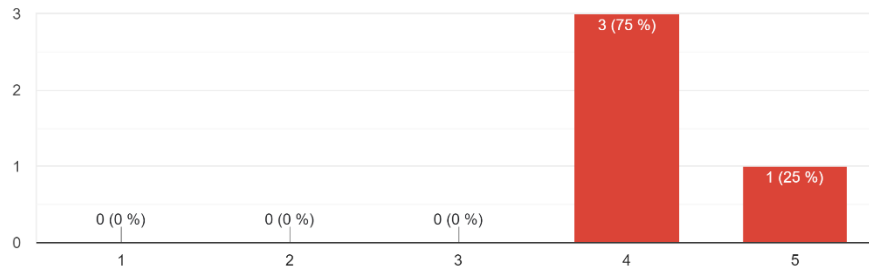


Ilustración 156. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 4.

Valora los tiempos de carga

4 respuestas

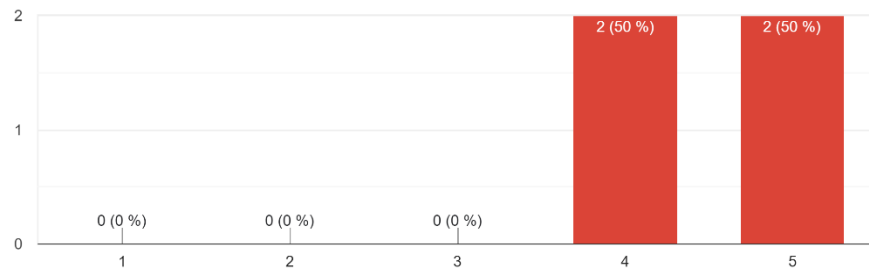


Ilustración 157. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 5.

Valora los colores de la interfaz

4 respuestas

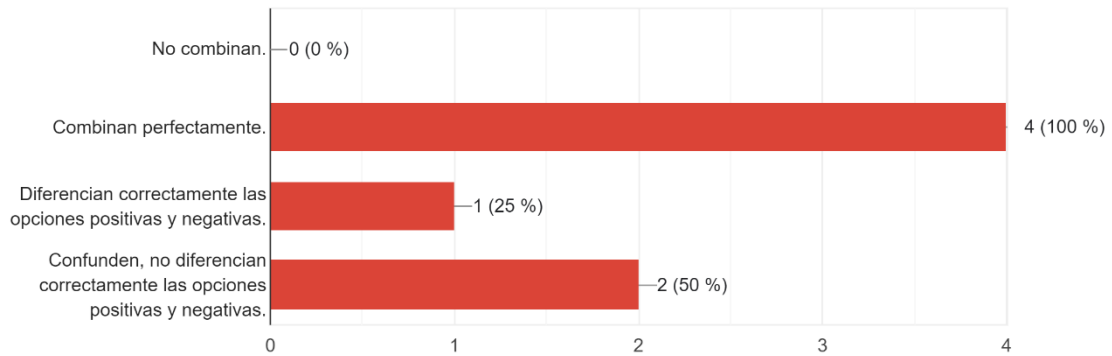


Ilustración 158. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 6.

Sugerencias
2 respuestas

La notificación de las actividades automáticas podía dar un poco de información a parte de la longitud y la latitud.

Pondría los nombres de las opciones en la barra de menú, no sólo al activarlos

Ilustración 159. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 7.

Opiniones
3 respuestas

Me gusta mucho la app.

Funciona muy bien

Me ha gustado y es útil.

Ilustración 160. Ejecución de las pruebas de usabilidad: Cuestiones de usabilidad 8.

6.5.1.4. Correcciones

Cómo se puede ver, los resultados de usabilidad son bastante buenos, pero a partir de estas pruebas se pueden realizar mejoras para aumentar la facilidad de uso de la aplicación por parte de los clientes.

Primero se modifica la notificación de las actividades automáticas, para que muestre información más relevante.

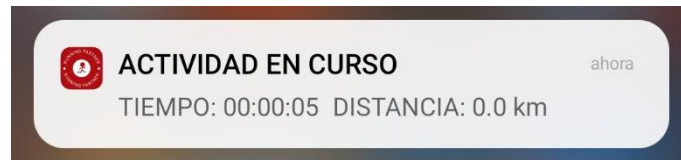


Ilustración 161. Corrección notificación.

Se modifican también algunos colores de los botones para que produzcan menos dudas.

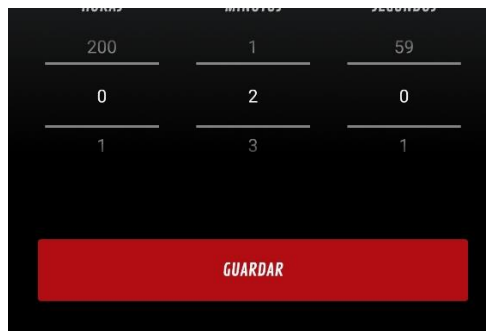


Ilustración 162. Corrección colores.

Finalmente, para que el usuario esté ubicado en todo momento, se hace visible continuamente el texto del menú inferior de la aplicación.



Ilustración 163. Corrección barra 1.



Ilustración 164. Corrección barra 2.

6.6. CSI 6: ELABORACIÓN DE LOS MANUALES DE USUARIO

En el siguiente apartado se especificarán los manuales de instalación, de usuario y de programador.

6.6.1. Manual de instalación

La instalación de esta aplicación se hará por medio de un APK, formato en el que se pueden encontrar muchas aplicaciones Android.

Como no se ha planeado abrirla al público general ni una comercialización, la manera de conseguir esta APK será generándola en Android Studio o recibéndola por parte de un tercero.

Una vez en posesión de la APK, habrá que abrirla con nuestro dispositivo Android. Es posible que se necesite previamente habilitar la instalación de orígenes desconocidos [41] en los ajustes de nuestro dispositivo, dado que no se está realizando la instalación desde GooglePlay [42]. Tras esto, al abrir el APK se preguntará al usuario si desea instalar la aplicación, si aceptamos la instalación comenzará y pasado un tiempo tendremos la aplicación en nuestro dispositivo lista para usarse.

6.6.2. Manual de ejecución

Para ejecutar la aplicación habrá que localizar su icono en el dispositivo y pulsar sobre él.

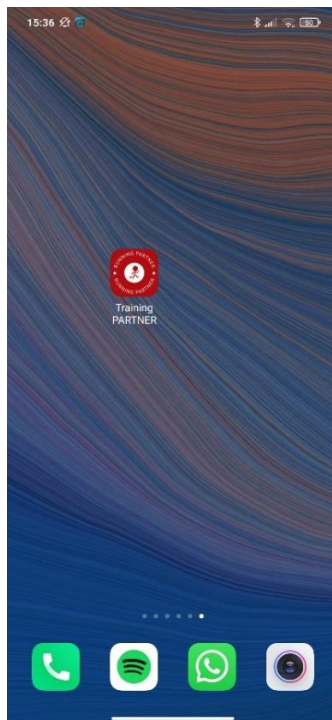


Ilustración 165. Icono de la aplicación.

6.6.3. Manual de Usuario

En este apartado se detallará el manual de Usuario, en el que se explicará al usuario final la manera de utilizar las distintas funcionalidades que ofrece la aplicación.

Para una mejor comprensión, las explicaciones se dividirán por funcionalidades y se usarán capturas de pantalla de la propia aplicación.

6.6.3.1. Registro e inicio de sesión

Una vez hayamos abierto e instalado la aplicación, la primera pantalla que se nos mostrará será la de registro, que servirá para crearnos credenciales en el sistema e iniciar sesión. En esta pantalla introduciremos nuestros datos personales que posteriormente se usarán para calcular métricas de salud o para identificar al usuario (paso 1).

En el momento en el que todos los campos estén completos, se pulsará el botón de Registrarse (paso 2). Si no hay errores de formato ni de credenciales, se registrará al usuario en el sistema y se guardarán sus credenciales para futuros accesos.



Ilustración 166. Manual de usuario: Registro.

En el caso de que el usuario ya esté en posesión de una cuenta dentro del sistema, podrá iniciar sesión, pulsando sobre el botón inferior. Tras esto se le redirigirá a la ventana de inicio de sesión.

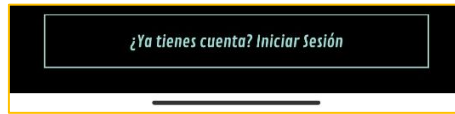


Ilustración 167. Manual de usuario: Botón iniciar sesión.

Para iniciar sesión en el sistema, el usuario introducirá su nombre de usuario y su contraseña (paso 1) y pulsará el botón de iniciar sesión. Si los datos son correctos se iniciará la sesión y se guardarán las credenciales.



Ilustración 168. Manual de usuario: Inicio de sesión.

Si se desea, se puede volver a la pantalla de registro pulsando el botón inferior.

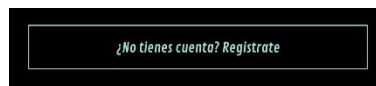


Ilustración 169: Manual de usuario: Botón de registro.

6.6.3.2. Cerrar sesión

Para cerrar sesión en la aplicación habrá que pulsar el botón superior derecho de la ventana principal (paso 1) y aceptar el mensaje emergente (paso 2).

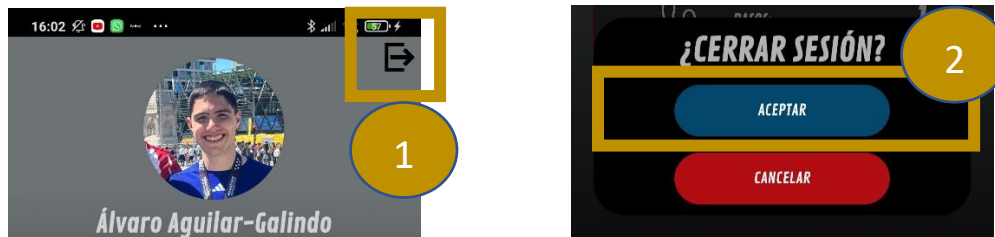


Ilustración 170. Manual de usuario: Cerrar sesión.

6.6.3.3. Cambiar foto de perfil

Para cambiar la foto de perfil habrá que pulsar sobre la imagen que se encuentra sobre el nombre del usuario.

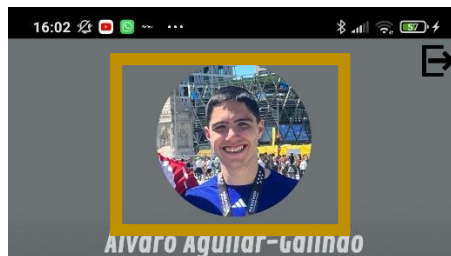


Ilustración 171. Manual de usuario: Cambiar foto de perfil 1.

Se abrirá una ventana en la que habrá que seleccionar la fuente de la que se quiere obtener la nueva imagen (paso 1). Tras esto se selecciona la imagen y el área que nos interesa y se pulsa el botón de cortar (paso 2).



Ilustración 172. Manual de usuario:
Cambiar foto de perfil 2.



Ilustración 173. Manual de usuario:
Cambiar foto de perfil 3.

6.6.3.4. Cambiar peso y altura

Para cambiar el peso y la altura habrá que pulsar sobre la imagen correspondiente (paso 1), e introducir el nuevo valor en el formato indicado (paso 2).

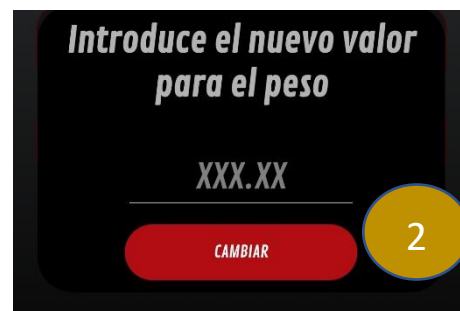


Ilustración 174. Manual de usuario: Cambiar peso y altura.

6.6.3.5. Grabar una actividad manual

Para grabar una actividad manual habrá que seleccionar el botón correspondiente en el menú inferior (paso 1), pulsar sobre *Manual* (paso 2), introducir los datos para que se active el botón (paso 3) y pulsar ese botón (paso 4).

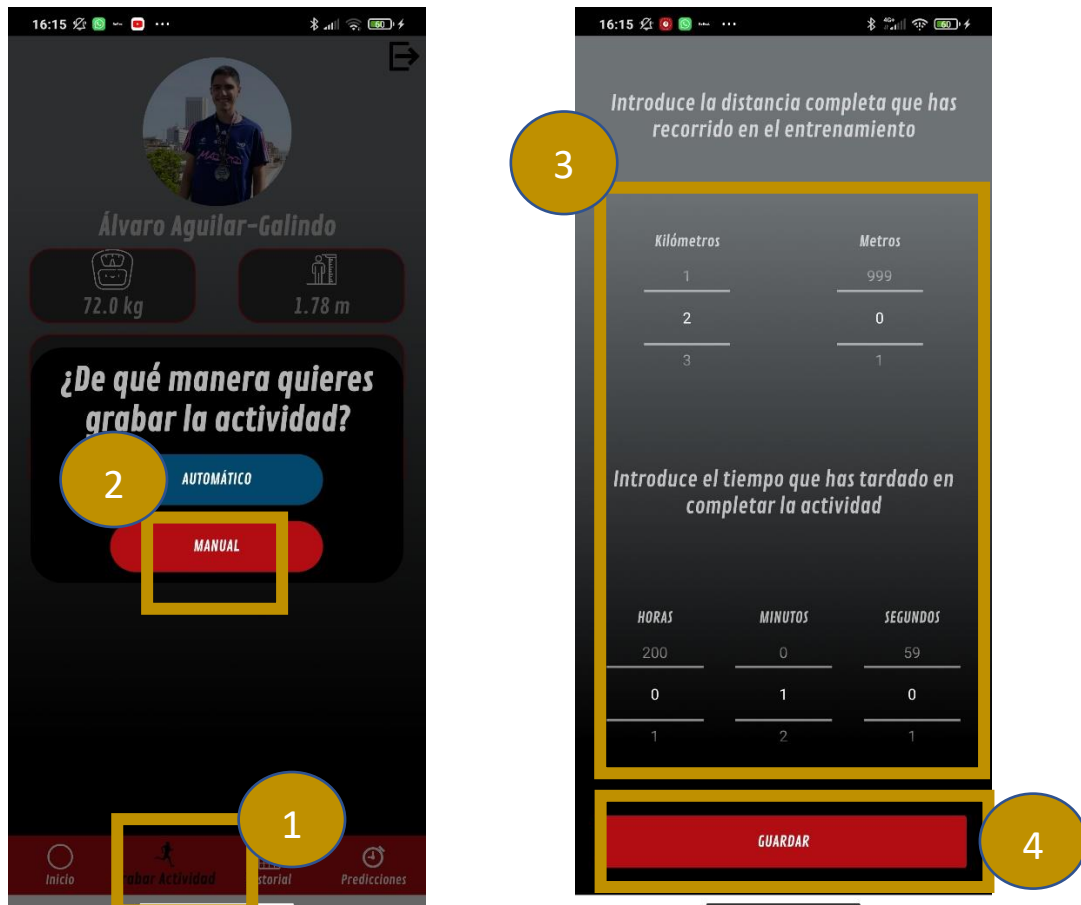


Ilustración 175. Manual de usuario: Grabar actividad manual.

6.6.3.6. Grabar una actividad con los sensores del dispositivo

Si es la primera vez que usamos esta funcionalidad, el sistema pedirá permisos para acceder a la ubicación del dispositivo. Para poder continuar habrá que aceptarla, en caso contrario el sistema redirigirá al usuario a la ventana principal y volverá a pedir permisos la próxima vez.

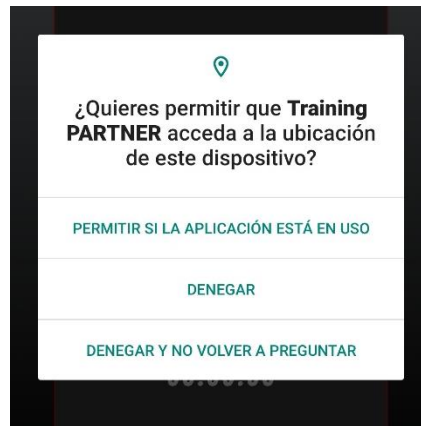


Ilustración 176. Manual de usuario: Permisos.

Es posible que en ocasiones la aplicación necesite un tiempo para acceder a la ubicación del dispositivo. Ocurre pocas veces, pero hay que destacar que la señal GPS siempre es mejor en exteriores o lugares abiertos. En estos casos se mostrará un mensaje al usuario que se eliminará en cuanto la ubicación haya sido establecida.



Ilustración 177. Manual de usuario: Determinando ubicación.

Una vez la ubicación haya sido establecida, se podrá comenzar la grabación de la actividad (paso 1). Tras esto se podrá pausar (paso 2) y terminar (paso 3) en cualquier momento.



Ilustración 178. Manual de usuario: Grabar actividades con los sensores del dispositivo.

Al finalizar la actividad, se mostrarán los detalles de esta.

6.6.3.7. Mostrar los detalles de una actividad

A los detalles de una actividad se accederá de dos maneras distintas:

- Tras realizar la actividad.
- Pulsando sobre uno de los elementos de la pantalla principal o de la pantalla de un día histórico.

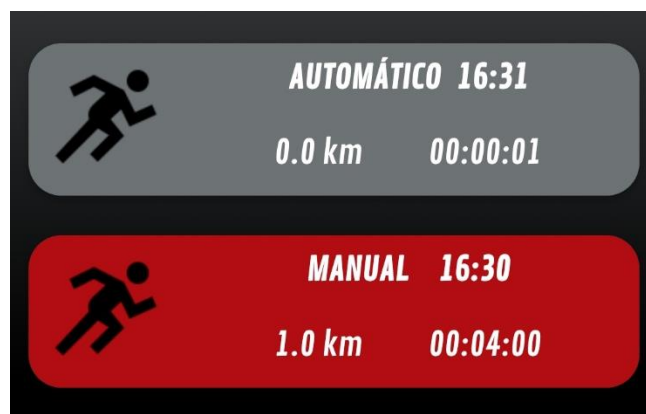


Ilustración 179. Manual de usuario: Elementos actividad.

En el caso de las actividades manuales sólo se podrán ver los datos, guardar la actividad (paso 1) o eliminarla (paso 2).

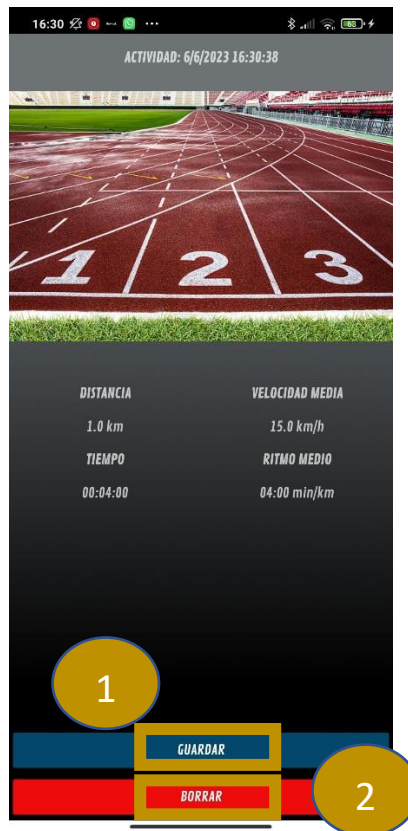


Ilustración 180. Manual de usuario: Detalles de una actividad manual.

Las actividades automáticas a parte de lo mencionado anteriormente proporcionan unos parciales de tiempo y un mapa interactivo con los puntos geográficos.

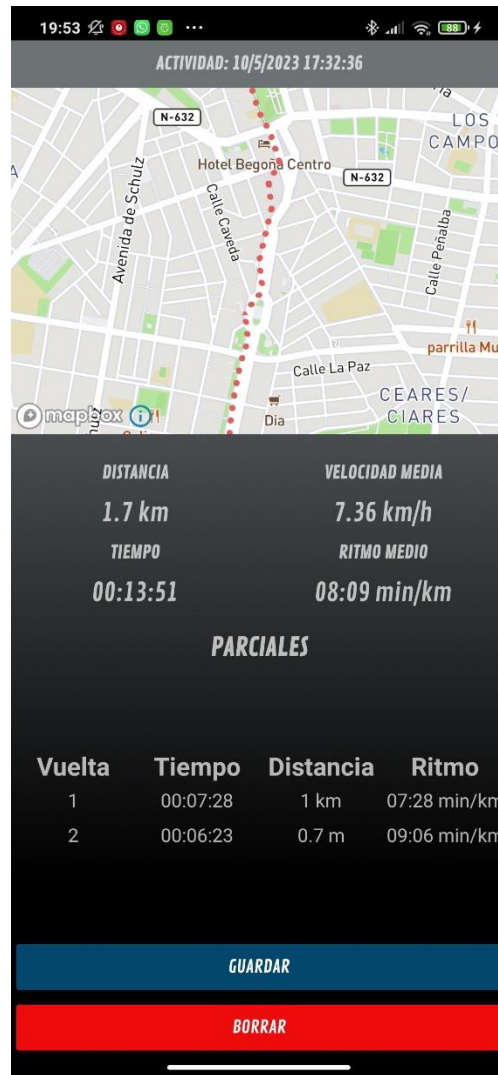


Ilustración 181. Manual de usuario: Detalles de una actividad automática.

6.6.3.8. Visualizar los pasos y calorías

Los pasos y calorías se calcularán automáticamente y se podrán visualizar en la pantalla principal.

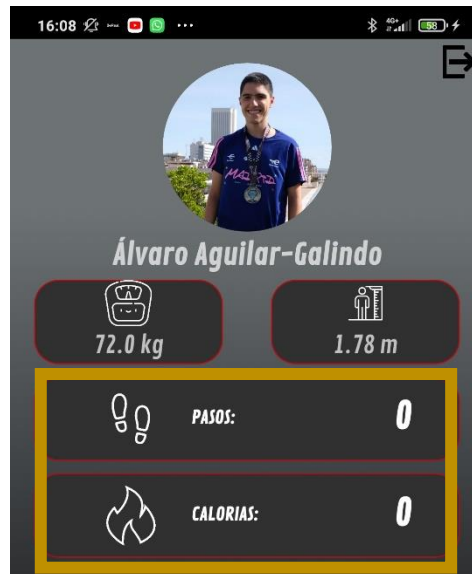


Ilustración 182. Manual de usuario: Visualización de pasos y calorías.

6.6.3.9. Visualización de un día histórico

Para acceder a un día pasado, habrá que pulsar sobre la opción de histórico (paso 1) en el menú inferior y acceder a el día que se desee (debe de estar activado), tras esto se pulsará el botón de visualizar (paso 2) y se accederán a los detalles de ese día concreto.

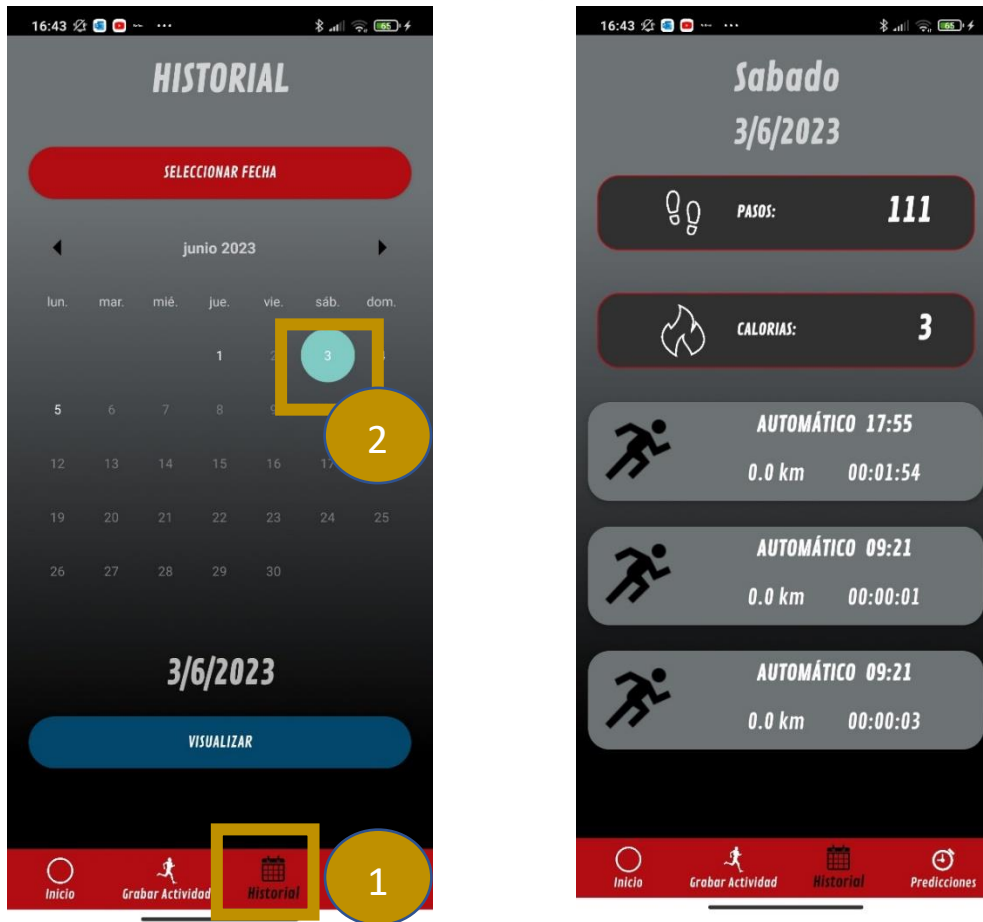


Ilustración 183. Manual de usuario: Visualización de un día histórico.

Es posible filtrar el mes que se muestra en el calendario en la pantalla de histórico, ya sea mediante el botón de seleccionar fecha o a través de las flechas de los laterales.



Ilustración 184. Manual de usuario: Seleccionar fecha.

6.6.3.10. Visualización de las predicciones

Para visualizar las predicciones simplemente habrá que pulsar el botón correspondiente en el menú inferior (paso 1). Si las condiciones se cumplen se mostrarán los tiempos, en caso contrario se desplegará un mensaje explicativo.

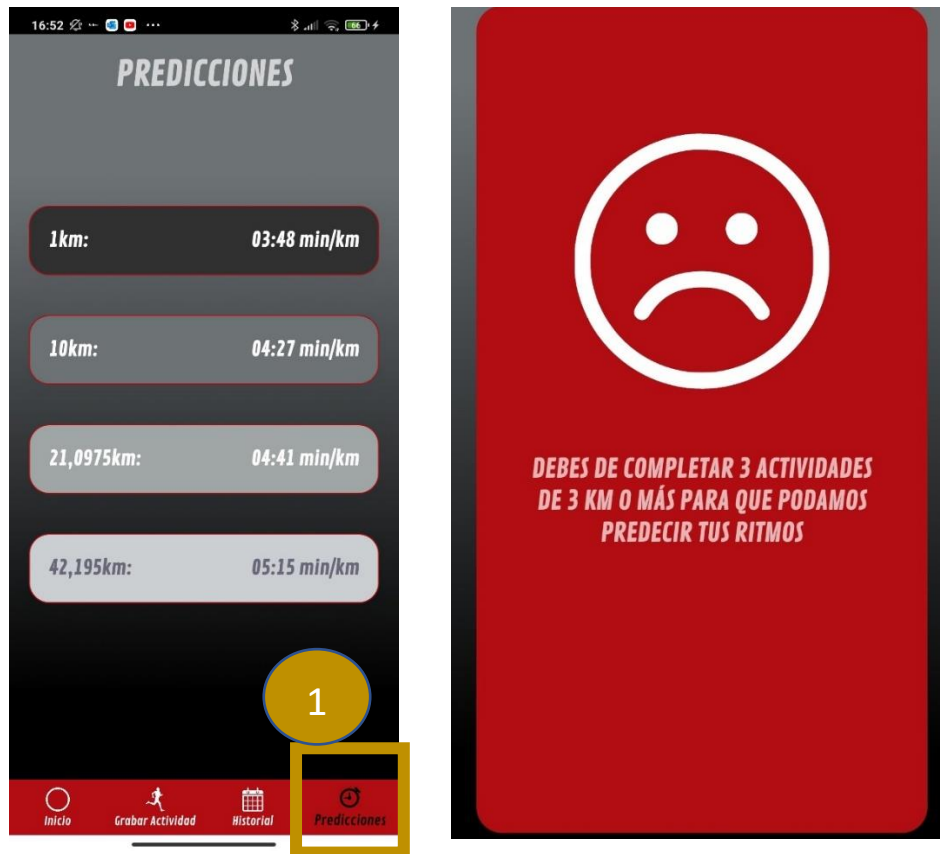


Ilustración 185. Manual de usuario: Visualización de las predicciones.

6.6.4. Manual del programador

6.6.4.1. *Android Manifest*

Este es un archivo que contiene todos los proyectos de Android Studio en el que se declara la información básica de la aplicación. Su tema, su icono, su nombre, las actividades, servicios y fragmentos que contiene, los permisos que necesita, etc.

6.6.4.2. *Archivo gradle*

Suele haber dos archivos gradle, uno a nivel de proyecto y otro a nivel de app. En ellos se especifica el sdk mínimo de la aplicación, el id de esta, el sdk al que está dirigido, la versión de java y todas las dependencias de la aplicación.

En el caso de este proyecto, nos encontramos con 4 tipos de dependencias:

- implementation: dependencias generales.
- testImplementation: dependencias para las pruebas.
- debugImplementation: dependencias para debug.
- androidTestImplementation: dependencias para los tests que tengan componentes Android.

```
implementation 'com.google.firebase:firebase-firestore:24.1.0'  
  
testImplementation 'junit:junit:4.12'  
testImplementation "androidx.arch.core:core-testing:2.1.0"  
debugImplementation 'androidx.test:runner:1.1.0'  
debugImplementation 'androidx.test:rules:1.1.0'  
debugImplementation 'androidx.test.ext:junit:1.1.3'  
androidTestImplementation 'androidx.test.espresso:espresso-core:3.4.0'  
debugImplementation 'androidx.fragment:fragment-testing:1.3.6'
```

Ilustración 186. Archivo gradle.

6.6.4.3. Ejecución de las pruebas

Para ejecutar las pruebas es necesario tener conectado un dispositivo Android desbloqueado al ordenador. Tras esto, sólo se necesita hacer click derecho sobre el fichero de prueba y ejecutarlo.

Si se desea ejecutar una prueba en concreto, bastará con ejecutar pulsar el símbolo de *play* verde situado a la izquierda del método.

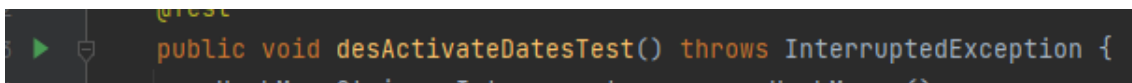


Ilustración 187. Ejecución de las pruebas.

6.6.4.4. Ejecución de la aplicación

Para ejecutar la aplicación en Android Studio, se debe de utilizar un dispositivo físico o virtual. En el caso de este proyecto, por falta de potencia del ordenador se ha ejecutado siempre la aplicación en un dispositivo físico Android.

El dispositivo ha de tener permisos de depuración, tras esto, aparecerá el modelo en Android studio y sólo será necesario ejecutar la app.



Ilustración 188. Ejecución de la aplicación.





CAPÍTULO 7: APÉNDICES

7.1. PLAN DE GESTIÓN DE RIESGOS

En este apartado se mostrará la manera de gestionar los riesgos identificados en [3.1.4. Riesgos](#).

7.1.1. Metodología

Para realizar la gestión de los riesgos se va a llevar a cabo una metodología específica, que describe el método y las fases por las que hay que pasar para realizar los trabajos relacionados con la Gestión de riesgos.

Boehm propuso la siguiente metodología:



Ilustración 189. Metodología 1.

La que se va a seguir en este proyecto es muy parecida a esta, pero añadiendo el paso previo de la planificación, de manera que se puedan decidir los criterios generales sobre los cuales se realizarán el resto de las fases.

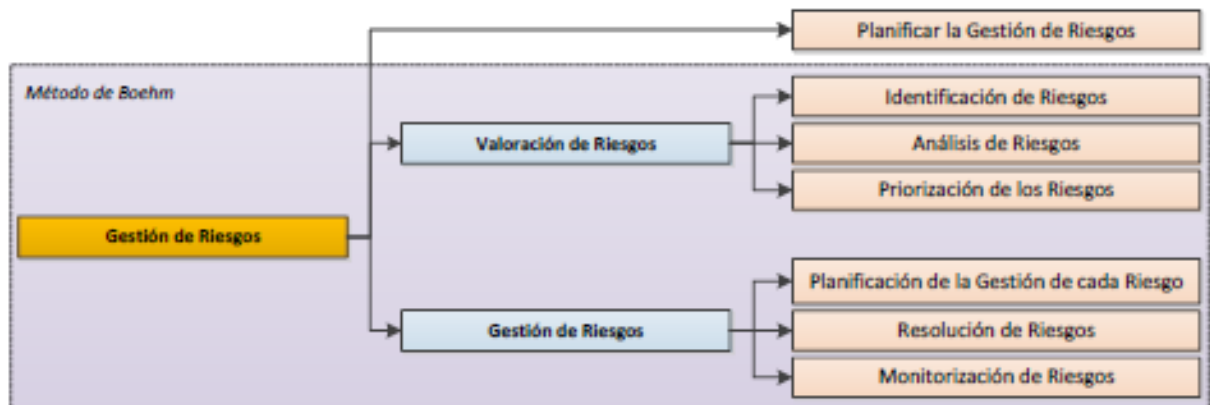


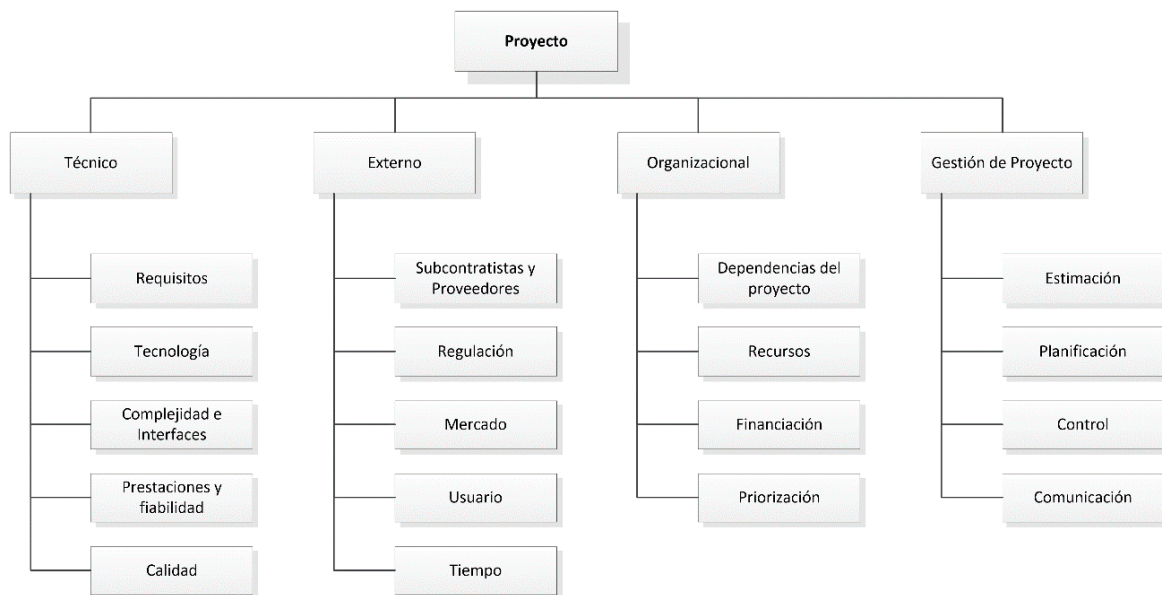
Ilustración 190. Metodología 2.

Esta metodología pasa entonces por 3 estados principales:

- Planificación de la Gestión de riesgos: Se deciden las políticas a aplicar en la gestión de riesgos.
- Valoración del riesgo: Estado en el que se identifican, analizan y priorizan los riesgos.
- Gestión de riesgos: En esta fase se planifica la gestión de cada riesgo, se resuelven y monitorizan.

7.1.2. Categorías de un riesgo

Para hacer más sencilla la identificación y manejo de los riesgos, se categorizan siguiendo un esquema similar al siguiente:



Esta estructura de desglose de riesgo (Risk Breakdown Structure o RBS) lista las categorías y subcategorías en las que los riesgos surgen para un proyecto típico. Dependiendo del tipo de proyecto y del tipo de organización, el RBS puede ser diferente. Una de las ventajas de este enfoque es que permite recordar a los participantes en un ejercicio de identificación de riesgos, las muchas fuentes de donde pueden surgir los riesgos del proyecto.

Ilustración 191. Categorías de un riesgo.

7.1.3. Probabilidad e impacto

La probabilidad de que un riesgo ocurra y el impacto que tiene este sobre el proyecto son valores que se usan para priorizar los riesgos.

Para realizar esta priorización se define la matriz de probabilidad e impacto:

Matriz de Probabilidad e Impacto										
Probabilidad	Amenazas					Oportunidades				
0,90	0,05	0,09	0,18	0,36	0,72	0,72	0,36	0,18	0,09	0,05
0,70	0,04	0,07	0,14	0,28	0,56	0,56	0,28	0,14	0,07	0,04
0,50	0,03	0,05	0,10	0,20	0,40	0,40	0,20	0,10	0,05	0,03
0,30	0,02	0,03	0,06	0,12	0,24	0,24	0,12	0,06	0,03	0,02
0,10	0,01	0,01	0,02	0,04	0,08	0,08	0,04	0,02	0,01	0,01
	0,05 Muy bajo	0,10 Bajo	0,20 Moderado	0,40 Alto	0,80 Muy Alto	0,80 Muy Alto	0,40 Alto	0,20 Moderado	0,10 Bajo	0,05 Muy Bajo
	Impacto Negativo					Impacto Positivo				

Cada riesgo es evaluado de acuerdo a la probabilidad de que ocurra y al impacto en algún objetivo si ocurriera. Los umbrales de tolerancia de cada organización se trasladan a la matriz, de manera que las áreas verde, amarilla y roja indiquen estos umbrales para la priorización de riesgos.

Ilustración 192. Matriz de probabilidad e impacto.

7.1.4. Estrategias frente a los riesgos

Se definen diferentes estrategias frente a los riesgos en el caso de que sean amenazas u oportunidades. Como en este proyecto sólo se han identificado riesgos que pueden ser considerados amenazas, se enumerarán las diferentes estrategias a realizar:

- Eliminación del riesgo: Se eliminan los factores que crean el riesgo, terminando así con la posibilidad de que ocurra.
- Transferencia del riesgo: Consiste en hacer que terceros (normalmente mediante contratos) asuman el riesgo, liberando así al equipo de su gestión.
- Evitación del riesgo: Identificar y controlar los factores que provocan el riesgo.
- Retención del riesgo: Consiste en asumir que el riesgo existe y que hay que convivir con él.

7.1.5. Identificación de los riesgos

La identificación de riesgos consiste en reflexionar a través de diferentes técnicas para encontrar los elementos que pueden suponer un riesgo para nuestro proyecto.

Esta acción se llevó a cabo en el apartado [3.1.4. Riesgos](#).

7.1.6. Lista de riesgos priorizada

A continuación, se muestran todos los riesgos identificados en el proyecto en una lista priorizada, en la que se describe también la estrategia a tomar y el modo de resolución para cada uno de ellos



ID	Nombre del Riesgo	Breve Descripción	Categoría	Probabilidad	Impacto				Impacto	Priorización	Respuesta al Riesgo	Estrategia
					Presup.	Planific.	Alcance	Calidad				
1	Poca experiencia con aplicaciones móviles.	El alumno ha cursado únicamente una asignatura referente a dispositivos móviles y de una manera muy teórica. La falta de conocimientos puede retrasar el proyecto.	Técnico - Complejidad	Muy Alta	Alto	Crítico	Alto	Alto	0,81		Hay que destinar un tiempo al estudio y aprendizaje de algunos conceptos que el alumno sólo conoce teóricamente, para así mejorar en sus habilidades desarrollando aplicaciones móviles.	Mitigar el riesgo
2	Poca experiencia con el desarrollo de pruebas.	El alumno nunca ha trabajado con las pruebas de espresso, además testear funcionalidades como la de grabación automática puede ser complejo. Esto puede retrasar la fecha de finalización del proyecto.	Técnico - Complejidad	Muy Alta	Medio	Alto	Medio	Alto	0,5		Habrà que planificar el proyecto teniendo en cuenta el tiempo que se puede tardar en desarrollar y ejecutar las pruebas.	Mitigar el riesgo
3	Falta de tiempo por el trabajo.	A la vez que se desarrolla el proyecto, el alumno está trabajando en una empresa a media jornada. La falta de tiempo puede retrasar el proyecto.	Técnico/Externo	Alta	Medio	Alto	Bajo	Medio	0,39		Llegar a un acuerdo con la empresa para obtener unos horarios que permitan trabajar correctamente en el proyecto.	Mitigar el riesgo



4	No tener muy clara la funcionalidad y requisitos deseados.	El tema de este proyecto ha sido propuesto por el alumno, si al inicio no se tiene muy clara la funcionalidad ni lo requisitos, se puede perder tiempo de trabajo.	Requisitos	Alta	Bajo	Alto	Bajo	Medio	0,39	Antes de comenzar a desarrollar el sistema hay que identificar el alcance y los requisitos, siguiendo un orden correcto.	Eliminar el riesgo
5	Perdida o mal funcionamiento del dispositivo.	El alumno sólo cuenta con un dispositivo Android en el que realizar las pruebas y ejecutar la aplicación. Dicho dispositivo es bastante antiguo y puede estropearse, lo cual desencadenaría un gasto económico y un retraso temporal.	Recursos	Media	Alto	Medio	Bajo	Inapreciable	0,28	Habrà que cuidar el correcto funcionamiento del dispositivo, incluso reservàndolo ùnicamente para probar el sistema y utilizar otro para uso personal si fuera necesario.	Mitigar el riesgo
6	Cambio en politica de pagos de alguna herramienta.	En el proyecto se utilizan dos herramientas de pago por uso (Google cloud y MapBox). Si las políticas de pago cambian, es posible que haya que modificar las herramientas, el uso que se haga de ellas o asumir un gasto económico que encarecería el presupuesto.	Recursos-Organizacional	Baja	Crítico	Medio	Inapreciable	Bajo	0,27	Realizar un seguimiento de las políticas de pagos de las herramientas para no recibir gastos inesperados. Habrà que contar con optativas por si el precio se encarece.	Mitigar el riesgo



7	Suspenseo de alguna asignatura.	Un suspenseo en alguna asignatura no permitiría al alumno entregar el proyecto en la fecha deseada.	Organizacional	Baja	Bajo	Alto	Medio	Bajo	0,17	No hay que descuidar el resto de las asignaturas independientemente de la importancia del proyecto.	Mitigar el riesgo
8	Mala climatología.	Al vivir en Asturias es probable que la climatología dificulte salir a probar la aplicación.	Externo	Muy Alta	Inapreciable	Bajo	Inapreciable	Bajo	0,14	En el caso de que la climatología no permita testear en el exterior el sistema, se pueden usar aplicaciones que falsean la ubicación del dispositivo.	Asumir el riesgo
9	Lesión deportiva.	Una lesión deportiva además de influir negativamente en el estado de ánimo del alumno puede dificultarle mucho el desarrollo de pruebas de la aplicación.	Externo	Baja	Inapreciable	Inapreciable	Inapreciable	Medio	0,09	En el caso que se de el riesgo hay que centrarse en recuperar de la lesión y pedir a algún conocido que pruebe la aplicación. Si es posible, también se puede testear andando o montando en bicicleta.	Asumir el riesgo
10	Baja de miembro del proyecto.	Este riesgo se da si el alumno o el profesor (jefe de proyecto) se ponen de baja, lo cual retrasaría o paralizaría el desarrollo del proyecto.	Externo-Recurso	Muy Baja	Bajo	Alto	Alto	Alto	0,06	Dependiendo de la duración y gravedad de la baja hay que plantearse la posibilidad de retrasar la entrega del proyecto.	Asumir el riesgo

Tabla 184. Lista de riesgos priorizada.



7.2. CONCLUSIONES

7.2.1. Conclusiones del proyecto

Se había planeado desarrollar una aplicación móvil que sirviera para tomar métricas de salud de los usuarios, tomar rutas de entrenamientos, guardar actividades de manera manual, permitir al usuario visualizar su histórico y crearles predicciones sobre los tiempos que serían capaces de realizar a partir de las actividades registradas.

Se perseguía desarrollar un subsistema para cada una de las funcionalidades y otro adicional para el registro e inicio de sesión. Se considera que todos los objetivos se han alcanzado de manera satisfactoria. Aunque en algunos casos como la toma de pasos o de puntos geográficos la precisión podría ser superior, se conocen las limitaciones del sistema y se ha identificado la manera de reducirlas en un futuro si se contase con más recursos.

El proyecto ha sido diseñado para que sea sencillo poder añadir nueva funcionalidad, o mejorar la actual. Además, se ha intentado realizar una aplicación usable y dirigida para todo tipo de público. Según los resultados de las pruebas de usabilidad, todo parece indicar que se ha logrado el objetivo.

Las decisiones tomadas al inicio se considera que han sido correctas pero mejorables. Es cierto que debido a la naturaleza del sistema es necesario realizar una aplicación móvil. Sin embargo, mientras se desarrollaba el proyecto se ha podido comprobar que tal vez Java actualmente no es el lenguaje idóneo para crear aplicaciones en Android Studio. Esto es debido a que existen alternativas como Kotlin, lenguaje del que no se disponía tanto dominio, pero que requiere de muchas menos líneas de código para programar las funcionalidades.

7.2.2. Conclusiones personales

A grandes rasgos me encuentro muy satisfecho con el resultado del proyecto realizado. Teniendo en cuenta que es la primera aplicación Android de *grandes* dimensiones que desarrollo, en mi opinión he creado un producto original y robusto.

He elegido un tema que me interesa y me gusta como es el deporte, más concretamente el running. Y a mi parecer ha sido un acierto, me he entretenido probándola y nunca me ha costado sentarme a desarrollar. Llevo investigando y documentando desde comienzos del curso académico, y aunque he de admitir que en muchas ocasiones ha sido un agobio, me parece que la organización ha sido muy buena. He procurado trabajar todos los días un poco, ya sea programando, documentando o investigando, y eso lo he notado en que no me he metido atracones de trabajo diarios ni semanales.

Al final del trabajo siento que he aprendido mucho más de lo que conocía al comienzo de este. He descubierto que me gusta desarrollar aplicaciones móviles, y me he formado en cómo crearlas. La labor de documentación, aunque ha sido la parte que menos atractiva me ha parecido, le veo un sentido práctico y organizativo. Al tener estructurada la documentación las fases de desarrollo y pruebas han sido mucho más sencillas y llevaderas.

7.3. AMPLIACIONES

A continuación, se detallan una serie de posibles ampliaciones que se podrían realizar al sistema.

7.3.1. Visión continua de un mapa durante las actividades

Esta fue una funcionalidad que se planteó al inicio del proyecto. Consistía en que, durante las actividades grabadas con los sensores del dispositivo, se pudiera ver constantemente el mapa y la ruta que se está trazando.

Esta idea se desechó en un inicio ya que se priorizó que la aplicación funcionara correctamente con la pantalla apagada. Tampoco se le veía mucho sentido al hecho de estar mirando el dispositivo mientras se hace ejercicio por la ciudad. Pero puede ser una ampliación interesante ya que es una funcionalidad práctica para las ocasiones en las que se desconoce el lugar en el que se está realizando la actividad.

7.3.2. Posibilidad de compartir las actividades

Sería interesante realizar una funcionalidad que permitiera crear una red de amigos que puedan visualizar nuestras actividades. De esta manera, se crearía un fragmento a parte en la aplicación en el que, además de poder ver los entrenamientos propios, se podrían visualizar los ajenos.

También habría la posibilidad de hacer una actividad privada, es decir que solo fuera visible por nosotros mismos. En un futuro incluso se podrían dar *likes* o escribir comentarios a los entrenamientos ajenos, de manera similar a la que lo hace Strava.



Ilustración 193. Posibilidad de compartir actividades.

7.3.3. Visión de actividades en dashboard web

La idea de esta ampliación sería permitir que las actividades y datos básicos del usuario pudieran verse en una web. Las funcionalidades de grabar actividades y cambiar datos quedarían excluidas solo para la aplicación móvil.

En esta web se podrían consultar los entrenamientos, los datos generales del usuario, el histórico y las predicciones. A futuro incluso se podrían añadir gráficas para representar mejor los datos.

Un buen ejemplo de esta funcionalidad es Runalyze [43], que toma los datos de aplicaciones de terceros como Garmin o Strava y las muestra en un servicio web.

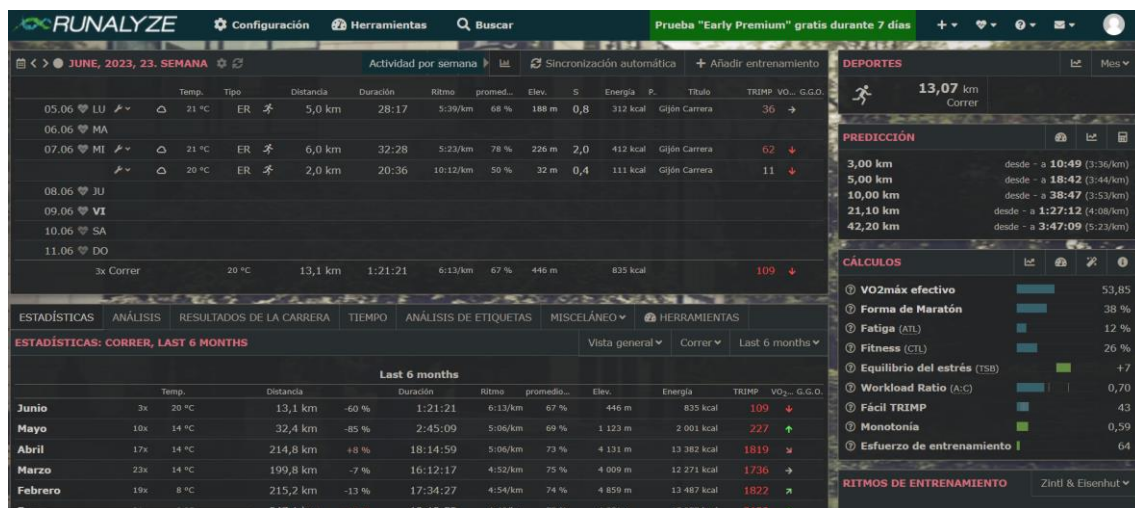


Ilustración 194. Dashboard web.

7.4. PROBLEMAS ENCONTRADOS

Durante el desarrollo del sistema se han encontrado gran cantidad de problemas, en este apartado se van a comentar los más importantes.

7.4.1. El servicio se paraba después de un tiempo en segundo plano

El problema que más quebraderos de cabeza causó fue que los sistemas de pasos y de geolocalización se paraban tras un tiempo en segundo plano. En un principio, para hacer que la aplicación se reactivara, se introdujo una notificación que mantenía el servicio activo. Sin embargo, tras un tiempo la notificación no se destruía, pero se dejaban de recibir actualizaciones tanto de pasos como de puntos geográficos.

El problema fue aún más difícil de resolver debido a que para probar la funcionalidad es necesario estar en movimiento, y depurar un servicio no es sencillo. Tras mucho tiempo pensando y buscando la solución, se encontró el problema. El fallo no venía por parte de la aplicación, si no del propio dispositivo. Era necesario quitar toda optimización de batería sobre la aplicación. El dispositivo mataba el servicio en segundo plano cuando llevaba un tiempo

ejecutándose. Me percaté que otras aplicaciones que acceden al sensor GPS del dispositivo también piden este ajuste, como es el caso de ZeppLife [44] de Xiaomi.

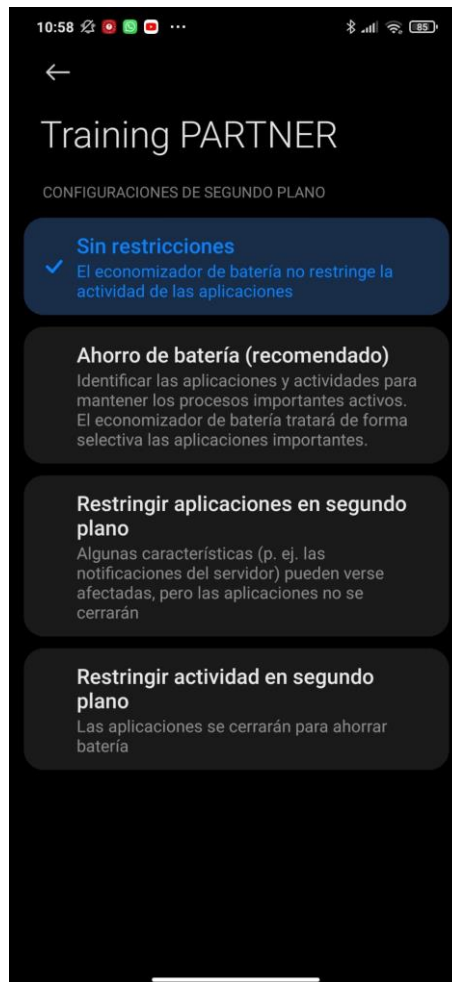


Ilustración 195. Sin restricciones de batería.

7.4.2. Manejar SharedPreferences

En un inicio se planeó el uso del fichero SharedPreferences para guardar los datos del usuario activo y los pasos. El problema vino en las pruebas, ya que muchos Activity y Fragments necesitaban ciertos datos en el fichero para ejecutarse de manera satisfactoria. Para suplir este problema, hubo que idear una manera de crear un Activity previo al que se quería probar. Este debía dar un contexto para poder editar el fichero SharedPreferences, y posteriormente inicializar el nuevo Activity, ya con el estado deseado.



7.5. REFERENCIAS BIBLIOGRÁFICAS

- [1] J. M. Redondo, ««Documentos-modelo para Trabajos de Fin de Grado/Master de la Escuela de Informática de Oviedo,»» 17 6 2019. [En línea]. Available: https://www.researchgate.net/publication/327882831_Plantilla_de_Proyectos_de_Fin_de_Carrera_de_la_Escuela_de_Informatica_de_Oviedo.
- [2] J. M. Redondo, «Creación y evaluación de plantillas para trabajos de fin de grado como buena práctica docente,» *Revista de Innovación y Buenas Prácticas Docentes*, p. pp, 2020. .
- [3] OMS, «Actividad física,» [En línea]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/physical-activity>. [Último acceso: 18 Diciembre 2022].
- [4] F. E. d. Corazón, «Hábitos de vida cardiosaludables,» [En línea]. Available: <https://fundaciondelcorazon.com/prensa/notas-de-prensa/1800-habitos-vida-cardiosaludables>. [Último acceso: 8 Enero 2023].
- [5] M. d. S. Español, «Encuesta Europea de Salud en España 2017,» [En línea]. Available: <https://www.sanidad.gob.es/estadEstudios/estadisticas/encuestaNacional/encuesta2017.html>. [Último acceso: 2022 Enero 19].
- [6] M. d. S. Español, «Encuesta Europea de Salud en España 2020,» [En línea]. Available: https://www.sanidad.gob.es/estadEstudios/estadisticas/EncuestaEuropea/EncuestaEuropea2020/EESE2020_inf_evol_princip_result.pdf. [Último acceso: 2022 Diciembre 20].
- [7] «Significado de Calorías (Qué son, Concepto y Definición),» [En línea]. Available: <https://www.significados.com/calorias/>. [Último acceso: 8 Enero 2023].
- [8] «Definición de acelerómetro- Qué es, significado y Concepto,» [En línea]. Available: <https://definicion.de/acelerometro/>. [Último acceso: 18 Diciembre 2022].
- [9] Xataka, «API: qué es y para qué sirve,» [En línea]. Available: <https://www.xataka.com/basics/api-que-sirve>. [Último acceso: 10 Enero 2023].
- [10] ABC, «Definición de Localización. Concepto en Definición ABC,» [En línea]. Available: <https://www.definicionabc.com/general/localizacion.php>. [Último acceso: 18 Enero 2023].
- [11] MapBox, «Mapbox: Maps, geocoding, and navigation APIs & SDKs,» [En línea]. Available: <https://www.mapbox.com/>. [Último acceso: 12 Enero 2023].
- [12] G. Developers, «Google Maps Platform - Google Developers,» [En línea]. Available: <https://developers.google.com/maps?hl=es-419>. [Último acceso: 12 Enero 2023].
- [13] «Mapbox vs Google Maps: ¿Qué API de mapas es mejor para su aplicación?,» [En línea]. Available: <https://geomatas.com/mapbox-vs-google->



maps/#:~:text=La%20ventaja%20de%20Mapbox%20frente,Mapbox%20s%C3%B3lo%20cobra%205%20d%C3%B3lares. [Último acceso: 20 Enero 2023].

- [14] A. Developers, «Funciones de Android Studio - Android Developers,» [En línea]. Available: <https://developer.android.com/studio/features?hl=es-419>. [Último acceso: 20 Enero 2023].
- [15] «Ventajas y desventajas de Java,» [En línea]. Available: <https://www.crehana.com/blog/transformacion-digital/ventajas-desventajas-java/>. [Último acceso: 20 Enero 2023].
- [16] A. F. Ocampo, «¿Los principales beneficios más importantes del desarrollo nativo de iOS con Swift,» [En línea]. Available: <https://andresfelipeocampo.medium.com/los-principales-beneficios-m%C3%A1s-importantes-del-desarrollo-nativo-de-ios-con-swift-866820163caf?> . [Último acceso: 25 Enero 2023].
- [17] Tribalyte, «C# (C Sharp): Qué es, dónde se utiliza y para qué sirve,» [En línea]. Available: <https://tech.tribalyte.eu/blog-c-sharp-que-es-para-que-sirve>. [Último acceso: 21 Enero 2023].
- [18] M. Learn, «¿Qué es Xamarin? - Xamarin,» [En línea]. Available: <https://learn.microsoft.com/es-es/xamarin/get-started/what-is-xamarin>. [Último acceso: 23 Enero 2023].
- [19] O. España, «¿Qué es una base de datos relacional?,» [En línea]. Available: <https://www.oracle.com/es/database/what-is-a-relational-database/>. [Último acceso: 25 Enero 2023].
- [20] S. Buzz, «SQLite: La Base de Datos Embebida,» [En línea]. Available: <https://sg.com.mx/revista/17/sqlite-la-base-datos-embebida>. [Último acceso: 25 Enero 2023].
- [21] dinahosting, «¿Qué es MySQL?,» [En línea]. Available: <https://dinahosting.com/ayuda/que-es-mysql/>. [Último acceso: 25 Enero 2023].
- [22] «Bases de datos documentales. ¿Qué es? Tipos y ejemplos,» [En línea]. Available: <https://ayudaleyprotecciondatos.es/bases-de-datos/documentales/>. [Último acceso: 30 Enero 2023].
- [23] OpenWebinars, «Qué es MongoDB y características,» [En línea]. Available: <https://openwebinars.net/blog/que-es-mongodb/>. [Último acceso: 2023 Enero 30].
- [24] «Firebase: qué es, para qué sirve, funcionalidades y ventajas,» [En línea]. Available: <https://digital55.com/blog/que-es-firebase-funcionalidades-ventajas-conclusiones/>. [Último acceso: 30 Enero 2023].
- [25] Microsoft, «Project,» [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365/project/project-management-software>. [Último acceso: 29 Mayo 2023].



- [26] Google, «Cloud,» [En línea]. Available: <https://cloud.google.com/?hl=es>.
- [27] «Glassdoor,» [En línea]. Available: <https://www.glassdoor.es/index.htm>.
- [28] Microsoft, «Visio,» [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365/visio/flowchart-software>. [Último acceso: 29 Mayo 2023].
- [29] «Free Logo Maker,» [En línea]. Available: <https://logo.com/>. [Último acceso: 29 Mayo 2023].
- [30] Oracle, «Java Code Conventions,» [En línea]. Available: <https://www.oracle.com/technetwork/java/codeconventions-150003.pdf>. [Último acceso: 28 Mayo 2023].
- [31] «MVC,» [En línea]. Available: <https://www.freecodecamp.org/espanol/news/el-modelo-de-arquitectura-view-controller-pattern/>. [Último acceso: 28 Mayo 2023].
- [32] Oracle, «Java,» [En línea]. Available: <https://www.java.com/es/>. [Último acceso: 28 Mayo 2023].
- [33] AWS, «XML,» [En línea]. Available: <https://aws.amazon.com/es/what-is/xml/>. [Último acceso: 28 Mayo 2023].
- [34] prolificinteractive, «material-calendarview,» [En línea]. Available: <https://github.com/prolificinteractive/material-calendarview>. [Último acceso: 29 Mayo 2023].
- [35] ArthurHub, «imageCrropper,» [En línea]. Available: <https://github.com/ArthurHub/Android-Image-Cropper>. [Último acceso: 28 Mayo 2023].
- [36] A. developers, «Espresso,» [En línea]. Available: <https://developer.android.com/training/testing/espresso?hl=es-419>. [Último acceso: 28 Mayo 2023].
- [37] A. developers, «Fragment-testing,» [En línea]. Available: <https://developer.android.com/guide/fragments/test?hl=es-419>. [Último acceso: 28 Mayo 2023].
- [38] A. developers, «Cloud Firestore,» [En línea]. Available: <https://firebase.google.com/docs/firestore?hl=es-419>. [Último acceso: 28 Mayo 2023].
- [39] Google, «Google storage,» [En línea]. Available: <https://cloud.google.com/storage?hl=es-419>. [Último acceso: 29 Mayo 2023].
- [40] Microsoft, «Word,» [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365/word>. [Último acceso: 29 Mayo 2023].
- [41] Microsoft, «Excel,» [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365/excel>. [Último acceso: 29 Mayo 2023].



- [42] Balsamiq, «Balsamiq,» [En línea]. Available: <https://balsamiq.com/>. [Último acceso: 29 Mayo 2023].
- [43] «Orígenes desconocidos,» [En línea]. Available: <https://espanol.verizon.com/support/knowledge-base-222177/>.
- [44] Google, «Google Play,» [En línea]. Available: <https://play.google.com/store/games?hl=es&gl=US&pli=1>.
- [45] Runalyze, «Runalyze,» [En línea]. Available: <https://runalyze.com/>.
- [46] Xiaomi, «ZeppLife,» [En línea]. Available: <https://play.google.com/store/apps/details?id=com.xiaomi.hm.health&hl=es&gl=US>.
- [47] Zoom, «Zoom,» [En línea]. Available: <https://zoom.us/>.
- [48] Vue, «Vue,» [En línea]. Available: <https://vuejs.org/>.
- [49] D. 2. G. O. Report, «Digital 2022 Global Overview Report,» [En línea]. Available: <https://datareportal.com/reports/digital-2022-global-overview-report>.
- [50] Microsoft, «Microsoft Suite,» [En línea]. Available: <https://www.microsoft.com/en-us/microsoft-365/products-apps-services>.
- [51] Github. [En línea]. Available: <https://github.com/>.
- [52] electronjs, «Electron,» [En línea]. Available: <https://www.electronjs.org/>.
- [53] ComputerHoy, «SOLID y PODS,» [En línea]. Available: <https://computerhoy.com/reportajes/tecnologia/proyecto-solid-que-es-504493>.
- [54] Angular, «Angular,» [En línea]. Available: <https://angular.io/>.
- [55] Amazon, «AWS,» [En línea]. Available: <https://aws.amazon.com/es/>.
- [56] «Slack,» [En línea]. Available: <https://slack.com/>.
- [57] «NodeJs,» [En línea]. Available: <https://nodejs.org/en>.
- [58] «Git,» [En línea]. Available: <https://git-scm.com/>.
- [59] «Digital 2022 Global Overview Report,» [En línea]. Available: <https://datareportal.com/reports/digital-2022-global-overview-report>.
- [60] «capacitor.js,» [En línea]. Available: <https://capacitorjs.com/>.
- [61] P. digital, «SOLID,» 15 Marzo 2023. [En línea]. Available: <https://www.paradigmadigital.com/dev/principios-solid-cuales-son-como-ayudan/>.
- [62] React, «React,» [En línea]. Available: <https://react.dev/learn/add-react-to-an-existing-project>.



-
- [63] Clavei, «B2C,» [En línea]. Available: <https://www.clavei.es/que-es/b2c/>.
- [64] Arimetrics, «B2B,» [En línea]. Available: <https://www.arimetrics.com/glosario-digital/b2b>.
- [65] TypeScript, «TypeScript,» [En línea]. Available: <https://www.typescriptlang.org/>.

7.6. CONTENIDOS ENTREGADOS EN LOS ANEXOS

7.6.1. Contenidos

Este trabajo de fin de grado será entregado en un fichero zip cuya estructura se detalla a continuación.

7.6.1.1. Estructura del archivo entregado

Ruta de acceso	Contenido
./	Contiene un archivo README.txt en el que se detalla la estructura del proyecto.
./trainingPartner_UO277955.zip	Zip que contiene todo el código de la aplicación desarrollada en este trabajo fin de grado.
./documentación	Carpeta con la documentación del trabajo en formato PDF.
./anexos	Carpeta con los anexos del trabajo. Su contenido se detallará a continuación.

Tabla 185. Estructura del archivo entregado.

7.6.1.2. Estructura del directorio de desarrollo

El directorio de desarrollo es ./trainingPartner_UO277955.zip.

Ruta de acceso	Contenido
./	Contiene dos carpetas app y gradle.
./gradle	Contiene los archivos gradle de la aplicación.
./app/src	Contiene el código fuente y las pruebas de la aplicación.
./app/src/main	Contiene el código de la aplicación.
./app/src/androidTest	Contiene los tests de la aplicación.

Tabla 186. Estructura del directorio de desarrollo.

7.6.1.3. Estructura del directorio de anexos

El directorio de desarrollo es ./anexos.

Ruta de acceso	Contenido
./riesgos	Carpeta que contiene los riesgos de la aplicación.
./presupuestos	Carpeta que contiene los presupuestos inicial y final de la aplicación en formato excel.
./planificaciones	Carpeta que contiene las planificaciones de la aplicación.
./trainingPartner.apk	Archivo APK de la aplicación.

Tabla 187. Estructura del directorio de anexos.



7.7. GNU FREE DOCUMENTATION LICENSE

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <<https://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

7.7.0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference.

7.7.1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.



The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties: any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License.



7.7.2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies.

7.7.3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest onto adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document.

7.7.4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this



License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.*
- B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.*
- C. State on the Title page the name of the publisher of the Modified Version, as the publisher.*
- D. Preserve all the copyright notices of the Document.*
- E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.*
- F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.*
- G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.*
- H. Include an unaltered copy of this License.*
- I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.*
- J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions it was based on. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.*
- K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.*
- L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.*
- M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version.*
- N. Do not retitle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section.*
- O. Preserve any Warranty Disclaimers.*



If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version.

7.7.5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements".

7.7.6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.



You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document.

7.7.7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate.

7.7.8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title.

7.7.9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.



Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it.

7.7.10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <https://www.gnu.org/licenses/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document.

7.7.11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.



The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing.

7.7.12. ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

```
Copyright (C) YEAR YOUR NAME.  
Permission is granted to copy, distribute and/or modify this  
document  
under the terms of the GNU Free Documentation License, Version 1.3  
or any later version published by the Free Software Foundation;  
with no Invariant Sections, no Front-Cover Texts, and no Back-  
Cover Texts.  
A copy of the license is included in the section entitled "GNU  
Free Documentation License".
```

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

```
with the Invariant Sections being LIST THEIR TITLES, with the  
Front-Cover Texts being LIST, and with the Back-Cover Texts being  
LIST.
```

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.