



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA
Escuela de Ingeniería Informática



Grado en Ingeniería Informática

Trabajo Fin de Grado

Sistema web para la monitorización de información de salud obtenida a través de dispositivos móviles

Juan Carlos Arroyo Herrera

Tutor: **Javier Sánchez Pérez**

Las Palmas de Gran Canaria
12 de julio de 2017

*"No digas: es imposible. Di: no lo he hecho todavía."
Proverbio japonés.*

Agradecimientos

Quiero agradecer a mi tutor Javier Sánchez por su ayuda y guía en este proyecto. A mi familia y mi pareja por su apoyo incondicional durante todo el tiempo de desarrollo. Y por último quiero agradecer a la población de StackOverFlow por su ayuda, pues sin ellos este proyecto no habría sido posible.

Resumen

El eHealth es un campo de la informática que se encuentra en auge, debido en primera instancia a su utilidad y segundo, a la aparición de hardware y software por parte de grandes empresas que lo llevan al público general. La mayoría de estas aplicaciones pecan por su falta de funcionalidades y de compatibilidad con distintos sistemas operativos y dispositivos.

El objetivo de este proyecto es la creación de la base de una aplicación web progresiva que abarque los sistemas operativos móviles más importantes como Android e iOS, así como se pueda utilizar en cualquier ordenador. Debido a las limitaciones el proyecto consistirá en una aplicación Android que recolecte los datos de una smartband por medio de Google Fit y que haciendo uso de Firebase se muestren los datos en una aplicación web. Además, la web permitirá añadir amigos, retarte a ti y a tus amigos y ver el historial de tus datos.

Para lograr realizar este proyecto se usan algunas de las tecnologías más punteras y actuales como Angular, Ionic, Sass y Nodejs. Además, para la planificación y desarrollo de este TFG y por el carácter especial del mismo se ha utilizado la metodología ágil SCRUM, así como, algunos aspectos de Running Lean.

Abstract

The eHealth is a field of computing that is booming, for the first time in its usefulness and second, an appearance of hardware and software by large companies that take it to the general public. Most of the applications of the sinned by its lack of functionalities and compatibility with different operating systems and devices.

The goal of this project is to create the basis of a progressive web application that covers the most important mobile operating systems like Android and iOS as well as can be used on any computer. Due to the limitations of the project will consist of an Android application that collects the data of an smartband by Google Fit and that the use of Firebase will measure the data in a web application. In addition, the web must add friends, challenge your friends and see the history of your data.

To achieve this project we use some of the latest and most advanced technologies such as Angular, Ionic, Sass and Nodejs. In addition, the SCRUM methodology has been used for the planning and development of this TFG and for the special character of it, as well as the few elements of Running Lean.

Índice general

1. Introducción	1
1.1. Objetivos	2
1.1.1. Objetivos generales del proyecto	2
1.1.2. Objetivos académicos	3
1.2. Aportaciones al entorno social	3
1.3. Estructura del documento	3
2. Estado actual del Arte	5
2.1. eHealth	5
2.2. Dispositivos móviles para eHealth	6
2.2.1. Xiaomi Mi Band 1s	6
2.2.2. Apple Watch	7
2.2.3. K'Watch Glucose	8
2.3. Software eHealth para dispositivos móviles	8
2.3.1. Google Fit	8
2.3.2. Samsung Health	9
2.3.3. Health	10
3. Recursos utilizados	13
3.1. Software	13
3.2. Librerías	16
3.3. Otros recursos utilizados	17
4. Plan de trabajo y temporización	19
4.1. Planificación del proyecto	19
4.1.1. Metodologías	19
4.2. Presupuesto	23
4.2.1. Costes Personal	24
4.2.2. Costes Hardware	24
4.2.3. Costes Software	25
4.2.4. Otros costes	26

ÍNDICE GENERAL

4.2.5.	Costes Totales	27
4.2.6.	Ingresos	28
5.	Desarrollo del proyecto	31
5.1.	Análisis	31
5.1.1.	Requisitos del Sistema	31
5.1.2.	Requisitos del Software	38
5.2.	Diseño	40
5.2.1.	Arquitectura	40
5.2.2.	Base de datos	40
5.2.3.	Diseño de las interfaces de usuario	41
5.2.4.	Diagramas de clases	44
5.3.	Implementación	45
5.3.1.	Java	45
5.3.2.	Android	46
5.3.3.	TypeScript	48
5.3.4.	Angular	49
5.3.5.	Ionic	50
5.3.6.	Firebase	51
5.3.7.	HTML5	53
5.3.8.	CSS3	54
5.3.9.	Sass	54
6.	Conclusiones y trabajo futuro	57
6.1.	Conclusiones	57
6.2.	Trabajo futuro	58
A.	Manual de usuario	59
A.1.	Aplicación Smartphone	59
A.2.	Aplicación web	62
A.2.1.	Inicio de sesión	62
A.2.2.	Inicio	63
A.2.3.	Amigos	65
A.2.4.	Búsqueda de amigos	66
A.2.5.	Retos	67
A.2.6.	Histórico	68
A.2.7.	Por intervalo de fechas	71
B.	Competencias específicas cubiertas	73

ÍNDICE GENERAL

C. Normativa y Legislación	75
C.1. Ley de Protección de Datos	75

ÍNDICE GENERAL

Índice de figuras

2.1. Xiaomi Mi Band 1S[48]	6
2.2. Apple Watch[7]	7
2.3. K'Watch Glucose[44]	8
2.4. Logo de Google Fit[15]	9
2.5. Logo de Samsung Health[39]	9
2.6. Logo de Health[8]	10
2.7. Logo de ResearchKit[10]	10
2.8. Logo de CareKit[9]	11
3.1. Logo de Sublime Text[40]	13
3.2. Logo de Node.js[12]	14
3.3. Logo de Android Studio[24]	14
3.4. Logo de TexStudio[43]	15
3.5. Logo de StarUML[35]	15
3.6. Logo de Balsamiq Mockups[11]	15
4.1. Figura que ilustra el ciclo de experimentación en Running Lean[38]	20
4.2. Figura que ilustra el modelo iterativo e incremental[1]	22
4.3. Figura que muestra el modelo de negocio en un Lean Canvas[38]	24
4.4. Figura que muestra la comparación de planes de Firebase.[20]	26
5.1. Primera figura que ilustra la pila de producto del proyecto	33
5.2. Segunda figura que ilustra la pila de producto del proyecto	34
5.3. Tercera figura que ilustra la pila de producto del proyecto	35
5.4. Cuarta figura que ilustra la pila de producto del proyecto	36
5.5. Quinta figura que ilustra la pila de producto del proyecto	37
5.6. Figura que ilustra las funcionalidades de la página web con casos de uso	39
5.7. Figura que ilustra las funcionalidades de la aplicación móvil con casos de uso	39

ÍNDICE DE FIGURAS

5.8. Figura que ilustra el flujo de información a través de los dispositivos y servicios	40
5.9. Figura que muestra la estructura de la base de datos	41
5.10. Figura que ilustra el diseño de la interfaz gráfica de la aplicación móvil	42
5.11. Figura que ilustra el diseño de la interfaz gráfica del perfil del usuario en la página web	43
5.12. Figura que ilustra el diseño de la interfaz gráfica de la página de amigos en la página web	43
5.13. Figura que ilustra el diagrama de clases de la aplicación móvil	44
5.14. Figura que ilustra el diagrama de clases de la página web . . .	45
5.15. Logo de Java[36]	46
5.16. Logo de Android[21]	47
5.17. Imagen que ilustra la fragmentación de las versiones de Android[2]	47
5.18. Código de muestra escrito en Typescript[2]	48
5.19. Logo de TypeScript[34]	48
5.20. Figura que muestra el patrón arquitectónico Modelo Vista Controlador (MVC)[45]	49
5.21. Figura que ilustra la arquitectura de Angular[4]	50
5.22. Logo de Angular[23]	50
5.23. Arquitectura de Ionic con Apache Cordova[18]	51
5.24. Logo de Ionic[19]	51
5.25. Logo de Firebase[32]	53
5.26. Logo de HTML5[47]	53
5.27. Logo de CSS[46]	54
5.28. Ejemplo código SASS[25]	55
5.29. Ejemplo código SCSS[26]	55
5.30. Logo de Sass[27]	56
A.1. Pantalla de inicio de sesión de la aplicación móvil	59
A.2. Pantalla principal que muestra los datos del usuario	60
A.3. Pantalla principal que muestra los datos del usuario y el botón de cerrar sesión	61
A.4. Página principal de inicio de sesión de la página web	62
A.5. Página de redireccionamiento del inicio de sesión de Google . .	63
A.6. Menú lateral de la página web	63
A.7. Primera parte de la página que muestra el perfil personal del usuario con sus datos	64
A.8. Segunda parte de la página que muestra el perfil personal del usuario con sus datos	65
A.9. Página que muestra a los amigos y sus datos y retos	65

ÍNDICE DE FIGURAS

A.10.Página que muestra el buscador de usuarios	66
A.11.Página que muestra la sección de retos de la aplicación web . .	67
A.12.Página que muestra el histórico de todos los datos para un día determinado	68
A.13.Página que muestra el histórico de pulsaciones para una hora determinada de un día	69
A.14.Página que muestra el histórico de glucosa para un rango de- terminado de horas de un día	70
A.15.Página que muestra gráficas de todos los datos diarios en un rango determinado de fechas	71
A.16.Página que muestra una gráfica del tipo de dato diario en un rango determinado de fechas	72

ÍNDICE DE FIGURAS

Índice de cuadros

4.1.	Tabla que muestra un resumen de las distintas fases del proyecto.	21
4.2.	Tabla que refleja los costes mensuales del personal.	24
4.3.	Tabla que refleja los costes del hardware por desarrollador. . .	25
4.4.	Tabla que refleja los costes del hardware para no desarrolladores.	25
4.5.	Tabla que refleja los costes mensuales de otros costes.	27
4.6.	Tabla que refleja los costes del material de oficina y mobiliario.	27
4.7.	Tabla que refleja los costes totales de mobiliario, hardware, etc.	27
4.8.	Tabla que refleja los costes anuales del proyecto.	28
4.9.	Tabla que refleja los costes acumulados aproximados de cinco años del proyecto.	29

ÍNDICE DE CUADROS

Capítulo 1

Introducción

Actualmente estamos presenciando la proliferación de la nueva moda de culto al cuerpo y la salud, después de años en los que se veía a un par de personas corriendo por las mañanas, sin embargo, ahora vemos a gente en bici, corriendo, haciendo ejercicio en el parque, estamos antes el nuevo boom de la salud y el ejercicio, por ello las empresas han empezado a desarrollar dispositivos que nos ayuden y permitan lograr nuestros objetivos y ayudarnos a llevar una vida más saludable. Por ello, nos encontramos multitud de dispositivos llamados wearables en los que podemos destacar pulseras y relojes inteligentes, zapatillas de deporte o incluso protectores bucales para boxeadores.

Estos dispositivos recogen datos por medio de multitud de sensores que generalmente se sincronizan con nuestro smartphone y nos muestran los datos como, por ejemplo, pulsaciones, glucosa en sangre, pasos, etc. Estos datos son útiles tanto para deportistas como para personas que solo buscan controlar su salud como en el caso de los diabéticos, por lo que existen varias aplicaciones de smartphone que aprovechan estos datos, además, estas se especializan en determinados sectores como, por ejemplo, la alimentación con aplicaciones para llevar el recuento de calorías, apps especiales para corredores, contador de repeticiones para el gimnasio, etc.

Sin embargo, aunque se están desarrollando grandes aplicaciones, en el mercado no se encuentran soluciones generales, debido en parte, a la gran segmentación producida por las marcas y la ausencia de estándares que hagan el trabajo más fácil a los desarrolladores.

Todo esto no es más que una parte del campo del eHealth, el cual es crucial para nuestra sociedad ya que busca facilitar, agilizar y abaratar la salud para ofrecer un mejor servicio. Por ello, la Unión Europea ha llevado a cabo iniciativas para favorecer la investigación y uso del campo[42]. Para ello, ha llevado a cabo un plan de acción entre el año 2004 y el 2011 que

buscaba cubrir las prescripciones médicas electrónicas y las tarjetas sanitarias electrónicas, con el fin de reducir los tiempos de espera, errores, etc. El segundo plan empezó en 2012 y estará activo hasta 2020, busca en primera instancia sentar las bases de los derechos de los pacientes y ayudar a la investigación, desarrollo e innovación del campo, además de facilitar la interoperabilidad de todos los servicios de eHealth.

Debido a lo anteriormente expuesto, este proyecto consistirá en la realización de un proyecto software desarrollado con SCRUM, que constará de una aplicación Android que recolecte los datos y de una aplicación web con algunas de las tecnologías más utilizadas, Angular e Ionic. Ambas aplicaciones conectadas mediante Firebase.

1.1. Objetivos

La motivación de este proyecto incide en la necesidad de sentar las bases de una plataforma que ayude a las personas a controlar su salud y así contribuir a nuestra sociedad.

A raíz de esto, podemos dividir los objetivos en dos grupos. El primero son los objetivos generales del proyecto, los cuales, son los que se buscan cumplir con el desarrollo de este proyecto. El segundo grupo son los objetivos académicos, que son aquellos que tienen como finalidad complementar mi formación, así como, adentrarse en algunas de las tecnologías más usadas en el mundo empresarial.

1.1.1. Objetivos generales del proyecto

Este proyecto tiene como finalidad utilizar los datos de un wearable y mostrarlos al usuario a la vez que este puede interactuar con sus amistades y competir con ellos. Por ello se ha elaborado la siguientes lista de objetivos.

- Programación de una página web de carácter social con algunos de los frameworks más usados.
- Utilizar una base de datos no relacional en tiempo real alojada en la nube como Firebase.
- Programación de una aplicación para Android que haga uso de la API de Google Fit.
- Utilización de datos de sensores de un wearable.

1.1.2. Objetivos académicos

Los objetivos académicos tienen la finalidad de complementar mi formación con la adquisición de conocimientos de diferentes herramientas ampliamente usadas y utilizar los conocimientos adquiridos en un proyecto real. A continuación, se muestran dichos objetivos.

- Aprender nuevas tecnologías como Angular, Ionic y Firebase.
- Utilizar los conocimientos adquiridos de ingeniería del software.
- Aplicar conocimientos de Scrum en la gestión de un proyecto real.

1.2. Aportaciones al entorno social

En los últimos años la informática se ha ido expandiendo a cada vez más sectores, uno de ellos es el eHealth, el cual está cogiendo fuerza debido a que diversos países están destinando recursos a informatizar su sistema de salud con proyectos como los historiales clínicos online. También las empresas innovan en productos tanto software como hardware para diversos campos del eHealth, como es en este caso, la monitorización de datos de salud de los usuarios por medio de sensores de dispositivos inteligentes, ya sean smartphones o wearables. Al centrarse este proyecto en la monitorización y además agregando un componente social, este puede contribuir al campo del eHealth ofreciendo un servicio que ayude a las personas a llevar una vida más sana, a deportistas a mejorar su condición física o a pacientes con enfermedades como la diabetes, que necesitan de un control de glucosa.

Por lo tanto, este proyecto puede servir como base de una aplicación que se pueda utilizar en cualquier smartphone u ordenador y recoja los datos de cualquier wearable, permitiendo que no solo los usuarios tengan acceso a los datos, sino que los padres vean los datos de sus hijos, el personal sanitario vea los datos de los pacientes o los datos de personas mayores.

1.3. Estructura del documento

A continuación, se expondrá un breve resumen de cada uno de los cinco capítulos en orden de aparición del documento. También se exponen los tres anexos del documento.

- **Capítulo 2. Estado del Arte:** en este capítulo se hablará del estado del eHealth, así como, de algunos dispositivos y software conocido o interesante.

- **Capítulo 3. Recursos utilizados:** en este apartado se exponen algunas de las herramientas, documentación, tecnologías, etc, usadas en el desarrollo del proyecto.
- **Capítulo 4. Plan de trabajo y temporización:** en este capítulo se expone la metodología usada, la organización del proyecto, así como, el presupuesto y plan de negocio del proyecto.
- **Capítulo 5. Desarrollo del proyecto:** este capítulo engloba todo el desarrollo del proyecto en todas sus fases, análisis, diseño e implementación. En estas secciones se mostrarán diagramas, prototipos de interfaces de usuario y se describirán en profundidad las tecnologías usadas más importantes del proyecto.
- **Capítulo 6. Conclusiones y trabajo futuro:** en este último capítulo se expondrá la conclusión del trabajo, así como, las posibles líneas de ampliación del proyecto.
- **Apéndice A. Manual de usuario:** en este anexo se muestran capturas de las aplicaciones, además, se describe como es su funcionamiento.
- **Apéndice B. Competencias específicas cubiertas:** este anexo contiene las competencias cubiertas en el desarrollo del proyecto.
- **Anexo. Normativa y legislación:** este último anexo expone las leyes que afectan al proyecto.

Capítulo 2

Estado actual del Arte

En este capítulo se expondrá el estado del eHealth en la sociedad, así como, algunos de los dispositivos más conocidos o más interesantes y se dará un repaso del software más interesante en el mercado.

2.1. eHealth

El eHealth es la fusión de la medicina con las tecnologías de la información y las comunicaciones. Esta se compone de diversos servicios o campos como, por ejemplo:

- **Historiales médicos electrónicos:** administración digital de historias clínicas que facilita la consulta e intercambio de datos de los pacientes entre los profesionales sanitarios.
- **Telemedicina:** este campo incluye todas las variantes de pruebas físicas y psicológicas que no requieren de la presencia del especialista.
- **Investigación médica usando grids** utiliza la computación en grid para manejar datos heterogéneos.
- **mHealth** es el campo que usa los dispositivos móviles como smartphones o smartbands para recolectar información de los pacientes y que estos sirvan tanto para los usuarios como para el personal sanitario.

Debido a la importancia de este campo, como se ha expuesto en la introducción, existen iniciativas por partes de los estados, la unión europea y diversas empresas como, por ejemplo, el Hackathon de salud celebrado en el pasado mes de junio en Madrid o Health4Good, una iniciativa de DKV.

Debido, a estas iniciativas y muchas otras, parece que el eHealth es un campo que gusta a las empresas y países por lo que se espera que se siga desarrollando, hasta al menos, una informatización casi total de la salud, sin embargo queda mucho camino que recorrer para ello.

2.2. Dispositivos móviles para eHealth

2.2.1. Xiaomi Mi Band 1s

La Xiaomi Mi Band 1S es una pulsera cuantificadora o smartband que pertenece al mundo de los wearables. Esta pulsera ha sido desarrollada por la empresa china Xiaomi. En poco tiempo esta smartband se ha puesto de moda debido a su bajo precio, menos de quince dólares, además de por las siguientes funcionalidades:

- Protección IP67.
- Treinta días de batería.
- Monitorización de pasos, calorías, distancia.
- Sesiones de ejercicio.
- Alarmas, notificaciones y control del sueño.
- Desbloquear el móvil.
- Monitor cardíaco.
- Aplicación para Android e iOS.



Figura 2.1: Xiaomi Mi Band 1S[48]

2.2.2. Apple Watch

El Watch de Apple es el único reloj inteligente de la compañía. Aunque su precio es elevado en comparación al resto de relojes inteligentes del mercado y de que solo es compatible con iPhone, goza de una cierta fama debido a su calidad de construcción y el software que incluye. A continuación se muestran algunas de sus características:

- GPS.
- Sensor de frecuencia cardíaca.
- Resistente al agua hasta 50 metros.
- Aplicación de entrenamiento con diferentes rutinas.
- Aplicación de actividad para contar los pasos, etc.
- Independiente del teléfono.
- Asistente virtual Siri.

Y muchas otras funcionalidades, lo que lo convierten en uno de los wearables de uso general más completos. También hay que destacar el compromiso de Apple con la salud y el gran sistema que ha creado, dando pie incluso a la posibilidad de medir la glucosa en sangre con el Watch.



Figura 2.2: Apple Watch[7]

2.2.3. K'Watch Glucose

El k'Watch Glucose es un smartwatch desarrollado por la empresa PK-vitality que actualmente se encuentra en fase de financiación colectiva y que promete darnos lecturas de nuestra glucosa en sangre de manera indolora. Este reloj tiene unas cargas denominadas K'apsul con treinta días de duración que son las encargadas de tomar la muestra de glucosa sin entrar en contacto con sangre, solamente con la piel.

Los datos se sincronizan con una aplicación dedicada, la cual nos permite compartir los datos con otras personas o profesionales, así como, mandar alertas a los usuarios. La aplicación está disponible tanto para Android como para iOS. Hay que destacar que la empresa también ha desarrollado un wearable similar llamado k'Watch Athlete que mide el ácido láctico.



Figura 2.3: K'Watch Glucose[44]

2.3. Software eHealth para dispositivos móviles

2.3.1. Google Fit

Google Fit es la aplicación de ejercicio de Google, esta se compone de una aplicación móvil para Android y además ofrece una aplicación web en ordenadores. Esta app recaba por medio de los sensores de los que disponemos, ya sea el propio teléfono o cualquier wearable una serie de datos, como

por ejemplos los pasos, la distancia o las calorías y nos permite crear retos personales. Además, establece sesiones automáticamente al caminar, correr o montar en bici, por ejemplo. También hay que destacar que Google Fit permite la sincronización de datos provenientes de otras aplicaciones como Mi Fit, Runkeeper o Nike+ y que también nos ofrece una API para que usemos los datos de la aplicación con permiso expreso del usuario.



Figura 2.4: Logo de Google Fit[15]

2.3.2. Samsung Health

Samsung Health es la aplicación de salud y deporte de la marca coreana, esta está disponible para dispositivos Android, pero solo permite la sincronización de datos con dispositivos Samsung como, por ejemplo, el Samsung Gear Fit 2. Aunque, también tiene acuerdos con otras empresas como Xiaomi o Garmin, las cuales producen sus propios dispositivos y te permiten compartir los datos entre la aplicación oficial y Samsung Health.

Esta aplicación nos permite, al igual que Google Fit, ver nuestros progresos diarios en retos, información como el sueño, pasos, distancia, etc. Además, nos ofrece la posibilidad de incluir lo que comemos para tener un historial de comidas y ver sus nutrientes, calorías etc. Esta funcionalidad al contrario que en Google Fit donde necesitas de aplicaciones externas, esta es parte de la app.



Figura 2.5: Logo de Samsung Health[39]

2.3.3. Health

Health es la aplicación de salud desarrollada por Apple para iOS y WatchOS. Al igual que la aplicación de salud de Samsung, esta da soporte al ejercicio, sueño, comida y otros datos de interés como los pasos y la distancia, sin embargo esta app no tiene un componente social como la aplicación de Samsung, sino que se centra más en el aspecto médico y de investigación puede analizar datos de salud reproductiva, tensión arterial, glucosa en sangre que el usuario introduce o que se obtienen de otros dispositivos.

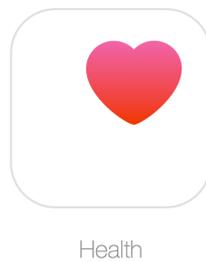


Figura 2.6: Logo de Health[8]

Hay que destacar que Apple está haciendo un trabajo formidable para la investigación médica, así como en el campo del eHealth con dos productos ResearchKit y CareKit.

ResearchKit es un framework de código libre utilizado para crear aplicaciones que puedan recopilar datos para investigaciones médicas con la previa autorización del usuario. Entre las aplicaciones más destacadas hechas con ResearchKit encontramos mPower una aplicación que hace uso del giroscopio y otras prestaciones del iPhone para realizar pruebas al usuario con Parkinson y así comprender mejor la enfermedad. Otro ejemplo es EpiWatch, una aplicación destinada a la investigación de los ataques epilépticos y como preverlos con el Watch, esa app consiste en la recopilación de datos en tiempo real, así como el aviso de familiares o médicos en cuanto el usuario sufre un ataque y pulsa una combinación de botones.



Figura 2.7: Logo de ResearchKit[10]

CareKit es un framework de software libre utilizado para crear aplicaciones para el uso personal y no de instituciones como ResearchKit. Este se centra en dar facilidades para desarrollar apps que ayuden a acercar a los pacientes con sus médicos o para llevar el control de su vida o de sus hijos. Una aplicación que está en desarrollo es "App para el cuidado de enfermedades crónicas" por el Beth Israel Deaconess Medical Center que permitirá al usuario ver la evolución de su enfermedad en el tiempo y comprobar que tratamientos funcionan mejor, así como compartirlos con sus médicos.

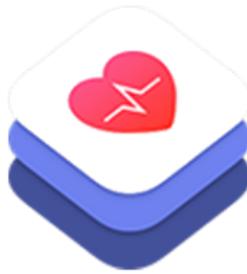


Figura 2.8: Logo de CareKit[9]

Capítulo 3

Recursos utilizados

A continuación se nombrarán los recursos utilizados en este TFG, algunos de ellos se explicarán o se dará la razón de su utilización.

3.1. Software

Sublime Text

Sublime Text es un editor de texto y editor de código fuente que permite la agregación de plugin de todo tipo de un modo bastante sencillo. Este software, aunque no es open source ni gratuito si permite su descargar y prueba indefinida. Este editor debido a su versatilidad y personalización es una herramienta formidable para el desarrollo y por esa razón he decidido utilizarlo para programar la página web.



Figura 3.1: Logo de Sublime Text[40]

Node.js

Node.js es un entorno en tiempo de ejecución multiplataforma, de código abierto, para la capa del servidor principalmente y además está basado en el

lenguaje ECMAScript. Fue creado con el enfoque de ser útil en la creación de programas de red altamente escalables como, por ejemplo, servidores web. Además, nodejs tiene un gestor de paquetes llamado NPM que facilita en gran medida la instalación de librerías, proyectos, etc.



Figura 3.2: Logo de Node.js[12]

Android Studio

Android Studio es el IDE oficial para el desarrollo en la plataforma Android. Este software está disponible en Windows, macOS y Linux y reemplaza a Eclipse como IDE oficial. Hay que destacar que Android Studio está basado en IntelliJ IDEA, un IDE para el desarrollo en Java de la empresa JetBrains.



Figura 3.3: Logo de Android Studio[24]

TexStudio

TexStudio es un editor de LaTeX multiplataforma y de código libre con una interfaz amigable parecida a Texmaker. Este editor nos ofrece un visor de PDF integrado, así como autocompletado, corrector ortográfico y otras características.

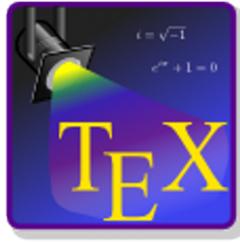


Figura 3.4: Logo de TexStudio[43]

StarUML

StarUML es una herramienta para el desarrollo con UML desarrollada por MKLab para Windows, macOS y Linux con una licencia GNU GPL. Con esta herramienta he realizado los distintos diagramas para el proyecto.



Figura 3.5: Logo de StarUML[35]

Balsamiq

Balsamiq Mockups es una aplicación de pago que nos permite realizar mockups de aplicaciones de cualquier tipo, web, móvil, etc. Esta aplicación facilita en gran medida la realización de diseños rápidos para la interfaz con el fin de demostrar conceptos o de hacerse una idea.



Figura 3.6: Logo de Balsamiq Mockups[11]

3.2. Librerías

- **AngularFire 2**[5] es la librería oficial para Firebase usando Angular 2. Nos permite utilizar la autenticación de Firebase y la base de datos en tiempo real por medio de los observables de la librería rxjs.
- **Ng2-charts**[17] es una librería basada en Chart.js para Angular 2. En el proyecto se usa para las gráficas de la página web.
- **Mydatepicker**[29] es un proyecto personal de Github que nos permite seleccionar una fecha con un calendario desplegable.
- **Mydaterangepicker**[30] al igual que el anterior es un proyecto personal de Github que nos permite seleccionar un rango de fechas con un calendario desplegable.
- **Angular-svg-round-progressbar**[31] se trata de una librería que nos permite usar una barra de progreso circular, la cual uso para mostrar el porcentaje de los retos.
- **Google Fit API** es la librería para deporte y salud de Google. De ella obtenemos ciertos datos como las calorías o los pasos, además nos sirve de interfaz común para un gran número de dispositivos y aplicaciones.
- **Google Sign In API** esta API desarrollada por el equipo de Google +, se usa para la autenticación centralizada en todos los servicios de Google, así como, para obtener los datos del usuario.

3.3. Otros recursos utilizados

- **Xiaomi Mi Band 1S** es el hardware utilizado para recoger algunos de los datos que maneja la web.
- **StackOverflow** es un conocido foro de dudas sobre programación.
- **LeanStack** es una página web que nos permite crear nuestro Lean Canvas gratuitamente.
- **Git** lo utilizo para gestionar las versiones del repositorio local del proyecto.
- **Tour of Heroes** [6] es un tutorial oficial de entrenamiento en Angular que nos va guiando a medida que vamos creando una aplicación y aprendiendo los distintos elementos del framework.
- **Libro Android 100%** [28] se trata de un libro gratuito que al igual que en el tutorial anterior guía al lector en la realización de diversos prototipos para que adquieras los conocimientos básicos de desarrollo en la plataforma.
- **Documentación de Ionic.** [16]
- **Documentación de Angular.** [3]
- **Documentación de Google Fit.** [14]
- **Documentación de Google Sign In.** [13]
- **Documentación de Firebase.** [22]

Capítulo 4

Plan de trabajo y temporización

En este capítulo se explicará la planificación del proyecto. Se abordará la metodología escogida y se realizará el presupuesto del proyecto.

4.1. Planificación del proyecto

4.1.1. Metodologías

En la elaboración de este proyecto se han utilizado dos metodologías. Scrum para el desarrollo y algún aspecto de Running Lean para el modelo de negocio y el presupuesto, esto es debido a las características del TFG por lo que no se ha podido aplicar Running Lean en su totalidad para verificar que el modelo de negocio tiene futuro, pero sería interesante contemplarlo de cara al futuro.

Running Lean

Running Lean[33] junto a Lean Startup [37] es una metodología que busca reducir al mínimo los desperdicios de nuestra StartUp, esta se nutre de diferentes metodologías y pensamientos. A continuación, se exponen los puntos que caracterizan a Running Lean.

- Prueba las ideas analizando la conducta del cliente.
- Implica a los clientes en todo el ciclo de desarrollo.
- Sigue un modelo iterativo con iteraciones cortas, además de validar el producto y el mercado paralelamente.

- Es un proceso disciplinado y riguroso.
- Es aprendizaje, rapidez y concentración

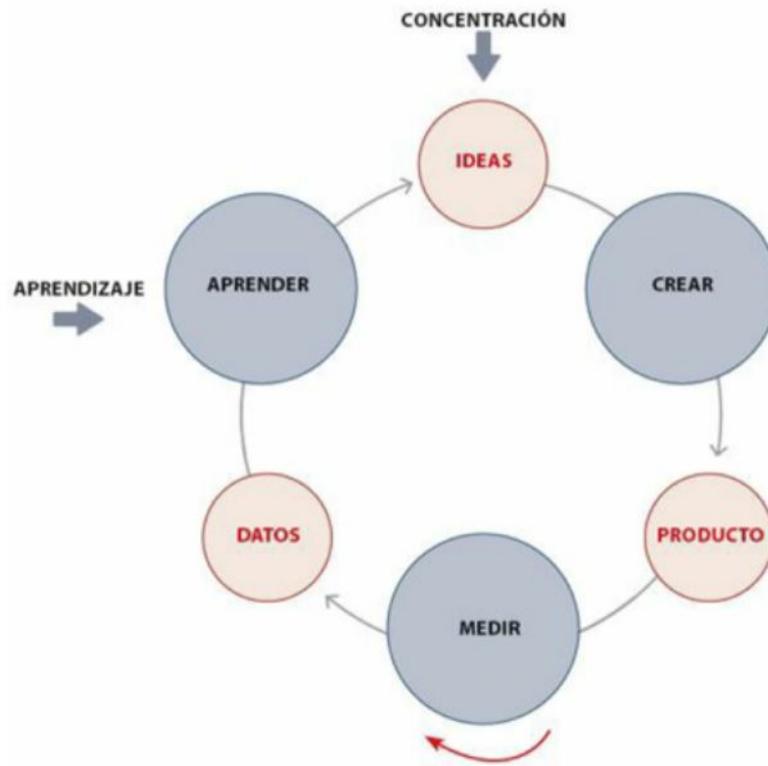


Figura 4.1: Figura que ilustra el ciclo de experimentación en Running Lean[38]

Este método se puede resumir en tres partes:

- Documenta tu plan: en esta parte debemos crear una serie de Lean Canvas con diferentes modelos de negocio para verificar que los posibles clientes tienen los problemas que nosotros queremos dar solución.
- Identifica los elementos de riesgo de tu plan: Running Lean busca minimizar los riesgos de una StartUp, como por ejemplo, crear un producto que nadie quiere utilizar. A su vez esta se divide en tres fases, ajuste problema/solución, ajuste producto/mercado y escala.
- Pon a prueba tu plan de manera sistemática: esto se consigue con la realización de entrevistas a clientes presentándoles en primer lugar un modelo de negocio en un Lean Canvas, a continuación una demo y al final una Landing Page que capte a los posibles clientes.

Scrum

Scrum es una metodología ágil que evoluciona en función de la organización y se adapta a nuestras necesidades. Este se basa al menos en un principio en un modelo iterativo mental que consiste en la iteración de varias fases como, por ejemplo, análisis, diseño, implementación y pruebas.

A continuación, se expone una tabla a modo de resumen donde podemos observar las distintas fases llevadas a cabo en este proyecto, además de una breve descripción de la actividad llevada en cada una de ellas así como, la estimación en horas del tiempo empleado.

Fases	Actividad	Horas
Análisis	Se analizará las necesidades y límites de las características del TFG.	20
Aprendizaje de las herramientas	Se procederá a estudiar todas las tecnologías y herramientas necesarias para el proyecto.	60
Diseño	Se llevará a cabo el diseño de los distintos componentes del proyecto, como el modelo o la interfaz.	80
Implementación	Se desarrollara el código que cumpla con lo establecido y utilizando un modelo lo más modular posible.	80
Pruebas	Se depurarán las funcionalidades del software.	20
Documentación	Se documentará el trabajo hecho	40

Cuadro 4.1: Tabla que muestra un resumen de las distintas fases del proyecto.

Este modelo nos permite realizar incrementos de funcionalidades terminadas, con el fin de obtener software que funciona y de valor en cada iteración. A continuación, se muestra una imagen ilustrativa.

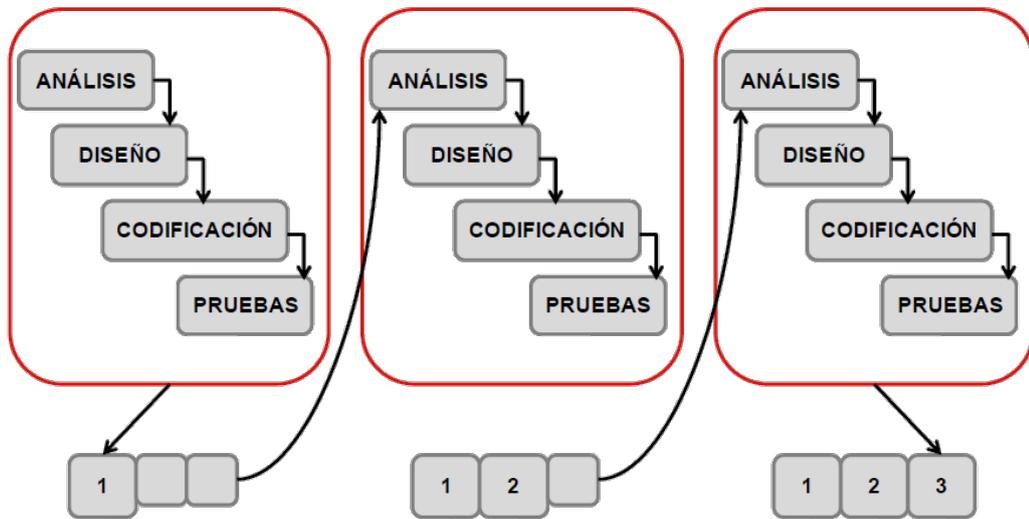


Figura 4.2: Figura que ilustra el modelo iterativo e incremental[1]

En Scrum, una vez se adquiere experiencia en el desarrollo, se sustituye el modelo incremental por integración continua lo que nos permite ir desarrollando funcionalidades y obteniendo feedback. Hay que destacar que no se usó Scrum como el marco técnico sino el Scrum avanzado, el cual se caracteriza por la utilización de ideas frente a procesos.

Por lo que para el desarrollo de este proyecto he utilizado algunas ideas, pero no todas debido a la imposibilidad de llevarlas a cabo como, por ejemplo, el hecho de que Scrum está orientada a equipos pequeños de personas. Para la planificación se describieron una serie de historias de usuario que podrían tener tanto la web como la aplicación móvil, sin importar si estas se iban a desarrollar o se quedarían como trabajo futuro. Las historias las podemos dividir en historias del usuario o del desarrollador en función del tipo de funcionalidad. A continuación, se estimaron las horas de las historias y se estableció una prioridad para cada una de ellas, algunas no se describieron debido a que eran historias épicas y con la prioridad baja, esto se debe a que el documento que recoge las historias de usuario es un documento vivo que evoluciona con el desarrollo, por lo tanto, se pueden dejar tareas que no estén en la cima de la pila de prioridad para el futuro.

Hemos organizado este proyecto en tres Sprints de aproximadamente un mes. Antes de empezar el Sprint se elegían las funcionalidades a desarrollar y al final se verificaban con el tutor. A continuación, se exponen las características de los tres Sprints.

- **Primer Sprint:** Este Sprint se centraba en los datos y en su obtención,

por lo que se diseñaría la base de datos y además se desarrollaría la aplicación Android.

- **Segundo Sprint:** En el segundo Sprint se daría comienzo a la página web, donde se desarrollarían las funcionalidades del perfil del usuario, la búsqueda de amigos y los amigos.
- **Tercer Sprint:** En el último Sprint se incluirían los gráficos para mostrar los datos, así como los retos y el histórico. También se afinarían algunos detalles del funcionamiento general y la apariencia.

Sin embargo, a medida que se fue avanzando en el desarrollo del proyecto se abandonaron los Sprints y se dio paso a la integración continua. Para la organización de las historias se usó un tablero Kanban, que inicialmente contenía las historias para cada Sprint, pero luego simplemente contenía todas las historias a desarrollar. Este tablero se usó debido a que aporta un apoyo visual de lo que queda por hacer, así como un impacto en la moral al dar por terminada una funcionalidad.

4.2. Presupuesto

Para la realización del presupuesto he utilizado un Lean Canvas para mostrar el modelo de negocio y además he incluido el desglose de los costes. A continuación, se muestra el Lean Canvas.

PROBLEM El usuario necesita monitorizar sus datos de salud El usuario quiere poder utilizar la plataforma en cualquier dispositivo El usuario quiere tener amigos en la plataforma El usuario quiere retarse a si mismo y a sus amigos EXISTING ALTERNATIVES Google Fit S Health Health (Apple)	SOLUTION Web + app Android/iOS/Windows Perfiles de amigos Retos Datos del usuario y su histórico	UNIQUE VALUE PROPOSITION Monitoriza tu salud cuando y donde quieras HIGH-LEVEL CONCEPT Una red social de la salud, el deporte y la alimentación.	UNFAIR ADVANTAGE Multiplataforma móvil y PC	CUSTOMER SEGMENTS Deportistas Personas con enfermedades que exigen monitorización con wearables Personas que practican deportes o ejercicio Personas que desean cuidar su salud EARLY ADOPTERS Personas diabéticas Padres que tienen que controlar la salud de sus hijos Personas que hacen ejercicio
	KEY METRICS Número de usuarios		CHANNELS Boca a boca	
COST STRUCTURE Costes de Firebase en función de la cantidad de usuarios Coste de personal bruto aproximado al mes 15000€ Coste de oficina 990€			REVENUE STREAMS Publicidad en las aplicaciones	

Figura 4.3: Figura que muestra el modelo de negocio en un Lean Canvas[38]

4.2.1. Costes Personal

En este apartado se muestran los costes del personal, todos ellos a jornada completa, excepto administración y mantenimiento que son a media jornada.

Personal	Coste(Bruto)
Scrum Manager	2500
Jefe de Proyecto	2500
Desarrollador	2500
Desarrollador	2500
Marketing	2500
Administración	1250
Mantenimiento	1250
Total	15.000

Cuadro 4.2: Tabla que refleja los costes mensuales del personal.

4.2.2. Costes Hardware

En esta sección se recoge los gastos de hardware dividido en dos grupos, los desarrolladores y los no desarrolladores como el jefe de proyecto o la administración.

Hardware	Coste
Ordenador de Sobremesa	800
Dos Monitores	200
Periféricos varios	80
Smartphone	500
Todos	1580

Cuadro 4.3: Tabla que refleja los costes del hardware por desarrollador.

Hardware	Coste
Ordenador de Sobremesa	800
Dos Monitores	200
Periféricos varios	80
Todos	1080

Cuadro 4.4: Tabla que refleja los costes del hardware para no desarrolladores.

4.2.3. Costes Software

Para calcular los costes de Software tenemos que saber que Firebase actualmente tiene tres planes. El primero es gratuito, pero solo nos sirve para el desarrollo de la aplicación y la página web. El segundo plan cuesta veinticinco dólares al mes que nos serviría para los primeros meses de vida de la aplicación, en función del crecimiento de usuarios. En el último plan se paga por el uso. En las siguientes imágenes vemos la comparación de los planes y sus precios. También hay que tener en cuenta que Sublime Text tiene un precio de setenta euros por licencia.

Productos	Plan Spark Límites generosos para aficionados Sin cargo	Plan Flame Precios predecibles para apps en expansión USD 25/mes	Plan Blaze Calcula los precios de las apps a gran escala Pago por uso
Incluido sin cargo Analytics, App Indexing, Dynamic Links, Invites, Notifications, Remote Config, Authentication, Cloud Messaging y Crash Reporting.	✓ Incluidos	✓ Incluidos	✓ Incluidos
Realtime Database Conexiones simultáneas ? GB almacenados GB descargados Copias de seguridad automáticas	100 1GB 10 GB/mes ✗	100K/instance 2.5 GB 20 GB/mes ✗	100K/instance \$5/GB USD 1/GB ✓
Storage ? GB almacenados GB descargados Operaciones de carga Operaciones de descarga	5 GB 1 GB/día 20,000/día 50,000/día	50 GB 50 GB/día 100K/day 250,000/día	USD 0.026/GB USD 0.12/GB USD 0.10/mil USD 0.01/mil
Cloud Functions ? Invocaciones GB-segundo CPU-segundo Redes de salida	125,000/mes 40,000/mes 40,000/mes Solo para Google	2,000,000/mes 400K/month 200K/month 5 GB/mes	USD 0.40/millón USD 0.0025/mil USD 0.01/mil USD 0.12/GB
Phone Auth ? US, Canada, India All other countries	10k/month	10k/month	\$0.01/verification \$0.06/verification
Hosting GB almacenados GB transferred Dominio personalizado y SSL	1 GB 10 GB/mes ✓	10 GB 50GB/month ✓	USD 0.026/GB USD 0.15/GB ✓
Test Lab ? Virtual Device Tests Physical Device Tests	10 tests/day 5 tests/day	10 tests/day 5 tests/day	\$1/device/hour \$5/device/hour

Figura 4.4: Figura que muestra la comparación de planes de Firebase.[20]

4.2.4. Otros costes

A continuación se expone una tabla que muestra los costes mensuales típicos de una oficina, alquiler, agua, luz, etc.

Tipo	Coste
Alquiler	800
Agua	20
Internet/Teléfono	60
Luz	60
Consumibles	50
Total	990

Cuadro 4.5: Tabla que refleja los costes mensuales de otros costes.

La siguiente tabla muestra los costes de adquisición de material de oficina, donde el mobiliario se compondría de mesas, sillas, pizarras, etc. En varios irían incluidos el material de limpieza o mantenimiento, teléfonos fijos, etc.

Tipo	Coste
Mobiliario	2000
Impresoras	200
Varios	150
Total	2350

Cuadro 4.6: Tabla que refleja los costes del material de oficina y mobiliario.

4.2.5. Costes Totales

En esta sección, se mostrarán los costes totales aproximados distribuidos en dos tablas, la primera contendrá los costes del material que se compra una vez y se renueva cada varios años o cuando se rompe. La segunda mostrará los costes anuales.

Tipo	Coste
Hardware	7980
Otros	2350
Software	210
Total	10.540

Cuadro 4.7: Tabla que refleja los costes totales de mobiliario, hardware, etc.

Tipo	Coste anual
Software	*
Personal	180.000
Otros Costes	11.880
Mantenimiento	36.000
Total	227.880

Cuadro 4.8: Tabla que refleja los costes anuales del proyecto.

* Valor dependiente del número de usuarios

4.2.6. Ingresos

Como se ha expuesto en el lean canvas anterior 4.3, los ingresos provendrían principalmente de la publicidad de las aplicaciones, aunque se podrían evaluar más opciones como las siguientes:

- Pago de la aplicación para eliminar la publicidad.
- Modelo freemium. Los usuarios pagarían por características especiales o premium.
- Pago por una suscripción anual o mensual.
- Pago único de la aplicación.

Sin embargo, el modelo escogido acarrea dificultades para empezar, puesto que necesita de una base de usuarios para generar ingresos ya que los ingresos medios por click en un anuncio son de entre uno y tres céntimos de euro. Por lo tanto, inicialmente se necesitarían otro tipo de ingresos como podría ser, la inversión de empresas o instituciones, la financiación colectiva por medio de un kickstarter, la promoción mediante una empresa deportiva o de salud como podría ser SportZone o incluso un préstamo bancario.

Debido a la dificultad al empezar se ha realizado una simulación de préstamo en el banco ING Direct. Este préstamo es en función de la facturación, en este caso de 227.880 euros, se nos daría un préstamo de 79.758 euros a pagar en cinco años con una cuota de 1503.21 euros al mes y unos gastos de apertura de 398.79. Esto nos ayudaría en los primeros meses en los que se adquiriera una base de usuarios que permita al proyecto seguir adelante por su propia cuenta. En la siguiente tabla vemos los costes totales del proyecto con el préstamo.

Costes	Primer año	Tercer año	Quinto año
Software	*	*	*
Personal	180.000	540.000	900.000
Otros Costes	11.880	35.640	59.400
Mantenimiento	36.000	108.000	180.000
Total	246.309	738.927	1.231.545

Cuadro 4.9: Tabla que refleja los costes acumulados aproximados de cinco años del proyecto.

* Valor dependiente del número de usuarios

Para calcular los ingresos propios de la aplicación de forma estimada tenemos que suponer un ingreso medio de dos céntimos por anuncio, un coste mensual de 15.990 euros y una media de 20 clicks por usuario, necesitaríamos de al menos 40.000 usuarios aproximadamente para cubrir el gasto mensual.

Capítulo 5

Desarrollo del proyecto

5.1. Análisis

5.1.1. Requisitos del Sistema

En esta sección se hablará de las limitaciones del proyecto y se expondrá la pila de producto.

Limitaciones del proyecto

A la hora de realizar este proyecto se tuvieron que tener una serie de limitaciones.

- **Limitaciones económicas:** no se podía disponer de todo el hardware que se deseara para la obtención de los datos y el desarrollo como, por ejemplo, un watch de Apple junto a un iPhone.
- **Limitación temporal:** este proyecto debía de realizarse en unas trescientas horas, una ardua tarea debido a la limitación de conocimientos.
- **Limitación de conocimientos:** esta limitación parte de que todas las tecnologías usadas en este proyecto de final de curso son desconocidas para el alumno, por lo que hubo un largo período de aprendizaje, así como, múltiples problemas que retrasaron el proyecto.

Pila de producto

A continuación, se muestra la Pila de Producto del proyecto recordando que esto es un documento vivo que cambia a lo largo del proyecto. Hay que tener en cuenta que no se desarrollaron todas las historias de usuario

mostradas y que estas representan un proyecto mayor que el que aquí se ha hecho debido a las limitaciones anteriores.

También, hay que destacar que al seguir Scrum no hace falta estimar todas las historias de golpe ni tampoco definir completamente, por lo que algunas son más completas que otras. Además, hay que señalar que la columna prioridad indica un nivel y no un número puesto que se prefiere expresar las más importantes, pero no elegir el orden. La columna de estimación expresa en horas aproximadas el tiempo que llevaría hacer cada historia teniendo en cuenta que incluye la interfaz gráfica como la lógica o los posibles cambios en la estructura de los datos. Por último, hay que destacar que las historias de usuario están escritas desde el punto de vista del usuario.

Nombre HU	Descripción	Razón / Resultado	Prioridad	Estimación
Sincronizar mis datos con Google Fit	necesito sincronizar mis datos con Google Fit	los datos de Google Fit se sincronizan con la aplicación	Alta	8
Iniciar sesión con mi cuenta de Google	quiero iniciar sesión con mi cuenta de Google	se inicia sesión con la cuenta de Google	Alta	8
Iniciar sesión con mi cuenta de	quiero iniciar sesión con mi cuenta de Facebook	se inicia sesión con la cuenta de Facebook	Alta	8
Ver perfil	quiero ver mi perfil	se muestra el perfil del usuario	Alta	8
Ver historial de pasos	quiero ver el historial de pasos	se muestra el historial de pasos diario/semanal/m	Alta	6
Ver historial de glucosa	quiero ver el historial de glucosa en sangre	se muestra el historial de la glucosa en sangre	Alta	8
Ver historial de pulsaciones	quiero ver el historial de mis pulsaciones	se muestran las pulsaciones diarias/semanales/mensuales	Alta	8
Ver historial de sueño	quiero ver el historial de sueño	se muestra el historial de sueño diario/semanal/m	Alta	6
Ver historial de distancia	quiero ver el historial de pasos	se muestra el historial de pasos diario/semanal/m	Alta	6
Ver historial de calorías	quiero ver el historial de pasos	se muestra el historial de pasos diario/semanal/m	Alta	6
Crear reto	quiero crear retos	para fomentar el ejercicio y el uso de la aplicación	Alta	8
Eliminar reto	quiero poder eliminar los retos	el reto queda eliminado	Alta	3
Ver progreso de los retos con un	quiero ver el progreso de los retos con un gráfico	se muestra el progreso de los retos con un gráfico	Alta	8
Ver datos diarios	quiero ver mis datos de cada día	se muestran los datos de ese día	Alta	7

Figura 5.1: Primera figura que ilustra la pila de producto del proyecto

Seguir a un amigo	quiero seguir a un amigo	para poder ver sus estadísticas	Alta	6
Barra de búsqueda de usuarios	quiero buscar a mis amigos	para poder seguirlos	Alta	6
Eliminar amigo	quiero eliminar a un amigo	se muestra la lista de amigos	Alta	3
Lista de amigos	quiero ver mi lista de	se muestra la lista de amigos	Alta	2
Ver retos de amigos	quiero ver el progreso de los retos de mis amigos	se muestran los retos de los amigos	Alta	8
Ver datos del día de mis amigos	quiero ver los datos del día de mis amigos	se muestran los datos diarios de los amigos	Alta	8
Ver datos en el tiempo con gráficas	quiero ver mis datos temporales en un rango de tiempo	se muestra una gráfica con datos como pulsaciones o glucosa	Alta	8
Crear Perfil con Google Sign In	quiero crear mi perfil	el perfil es creado	Alta	6
Crear Perfil con	quiero crear mi perfil	el perfil es creado	Alta	6
Modificar Perfil	quiero modificar mi perfil	el perfil es modificado	Alta	3
Ver estadísticas personales	quiero ver mis estadísticas	se muestran estadísticas de los datos día/mes/año	Alta	7
Sincronizar mi SmartBand	quiero sincronizar mi Smartband con la	para poder ver mis datos	Alta	
Sincronizar mi SmartWatch	quiero sincronizar mi Smartwach	para poder ver mis datos	Alta	
Sincronizar datos con Health	quiero sincronizar mis datos con la aplicación Health de Apple	se sincronizan los datos de Health	Alta	
Sincronizar datos con Samsung Health	quiero sincronizar mis datos con Samsung Health	se sincronizan los datos de Samsung Health	Alta	

Figura 5.2: Segunda figura que ilustra la pila de producto del proyecto

Sincronizar datos con Lifelog	quiero sincroniar mis datos con Lifelog	se sincronizan los datos de Lifelog	Alta	
Establecer cuenta de menor	quiero establecer la cuenta de un menor a mi cargo	se establece una cuenta de un menor ligada a la cuenta del tutor	Alta	
Ver datos de personas a tu cargo	quiero ver los datos personales de las personas a mi	se muestran los datos de las personas a tu cargo	Alta	
Eliminar Perfil	quiero eliminar mi perfil	el perfil es eliminado	Alta	
Ver perfil de amigo	quiero ver el perfil de un amigo	se muestra el perfil de un amigo	Alta	
Notificaciones de retos	quiero que me notifiquen los retos	para poder realizarlos	Alto	
Crear Perfil con Facebook	quiero crear mi perfil	el perfil es creado	Media	
Añadir comida en calorías	quiero añadir la cantidad de calorías de una	se añade una comida en calorías	Media	
Ver historial de peso	quiero ver mi historial de peso	se muestra el historial de peso	Media	
Establecer límite de calorías	quiero establecer un límite de calorías	se establece un límite de calorías	Media	
Eliminar límite de calorías	quiero eliminar el límite de calorías	se elimina el límite de calorías	Media	
Modificar límite de calorías	quiero modificar el límite de calorías	se modifica el límite de calorías	Media	
Modificar reto	quiero poder modificar los retos	el reto queda modificado	Media	
Establecer meta de peso	quiero establecer una meta de peso	se establece una meta de peso	Media	
Ver alarmas	quiero ver las alarmas establecidas	se muestran las alarmas	Media	

Figura 5.3: Tercera figura que ilustra la pila de producto del proyecto

Despertar automático	quiero que me despierten automáticamente cuando haya dormido un número de	se usa el wearable para despertar al usuario	Media	
Alertas de amistades	quiero que me notifiquen lo que hacen mis amistades	la aplicacion notifica logros de amigos	Media	
Activar/desactivar notificaciones	quiero poder silenciar o activar las notificaciones	se activan/desactivan las notificaciones	Media	
Iniciar actividad preestablecida	quiero iniciar una actividad preestablecida	se inicia una actividad preestablecida	Media	
Detener actividad preestablecida	quiero detener una actividad preestablecida	se detiene una actividad preestablecida	Media	
Iniciar actividad personalizada	quiero iniciar una actividad personalizada	se inicia una actividad personalizada	Media	
Crear actividad personalizada	quiero crear una actividad personalizada	se crea una actividad personalizada	Media	
Modificar actividad personalizada	quiero modificar una actividad personalizada	se modifica una actividad personalizada	Media	
Eliminar actividad personalizada	quiero eliminar una actividad personalizada	se elimina una actividad personalizada	Media	
Crear aviso personalizado	quiero crear un aviso personalizado	se crea un aviso personalizado	Media	
Modificar aviso personalizado	quiero modificar un aviso personalizado	se modifica un aviso personalizado	Media	
Eliminar aviso personalizado	quiero eliminar un aviso personalizado	se elimina un aviso personalizado	Media	
Aviso de pulsaciones extremas	quiero que me avisen cuando las pulsaciones son extremas	se notifica que las pulsaciones son extremas	Media	

Figura 5.4: Cuarta figura que ilustra la pila de producto del proyecto

Aviso de emergencia con ubicación	quiero poder mandar mi ubicación en caso de emergencia	se manda un aviso a los servicios de urgencias	Media	
Aviso de hipo/hiperglucemia	quiero que me avisen cuando tenga hipo/hiperglucemia	se notifica de hipo/hiperglucemia	Media	
Aviso de presión arterial alta/baja	quiero que me avisen cuando tenga la presión arterial demasiado	se notifica de presión arterial alta/baja	Media	
Modificar comida en calorías	quiero modificar la cantidad de calorías de la comida	se modifica la cantidad de calorías de una comida	Media	
Eliminar comida en calorías	quiero eliminar la cantidad de calorías de la comida	se elimina la cantidad de calorías de una comida	Media	
Sección de ayuda	quiero poder solucionar mis dudas con el uso de la aplicación	para poder resolver mis dudas sin tener que recurrir a atención al cliente	Baja	
Reportar Bug	quiero poder reportar fallos	para que se puedan	Baja	
Establecer alarma	quiero establecer una alarma	se establece una alarma	Baja	
Modificar alarma	quiero modificar una alarma establecida	se modifica la alarma	Baja	
Eliminar alarma	quiero eliminar una alarma	se elimina la alarma	Baja	
Establece etiqueta de alarma	quiero establecer una etiqueta a la alarma	se establece una etiqueta en la alarma seleccionada	Baja	
Sugerencias de ejercicio al usuario	quiero recibir sugerencias de ejercicio	se muestra una notificación de sugerencia de ejercicio	Bajo	

Figura 5.5: Quinta figura que ilustra la pila de producto del proyecto

5.1.2. Requisitos del Software

Los usuarios a los que va dirigido este proyecto son a todas las personas que deseen llevar una vida más sana mediante el uso de la monitorización de salud, personas que practiquen deportes o hagan ejercicio, así como personas con diferentes patologías que les exigen tener control de las mismas, así como, padres o tutores que deseen conocer en todo momento el estado de sus hijos o al revés si se trata de gente mayor.

Por último, también habría que tener en cuenta que la comunidad médica también podría estar interesada, ya que podrían facilitarles información relevante de la salud del paciente y ofrecer un mejor servicio.

Casos de Uso

El proyecto debe constar tanto de una aplicación móvil como de una página web, sin embargo, por las limitaciones descritas con anterioridad ha sido imposible crear una solución global que satisfaga todas las necesidades, por lo que se decidió desarrollar una aplicación Android básica y una página web donde estuviera el peso de las funcionalidades, todo ello teniendo en cuenta que es un prototipo.

A continuación, se exponen dos diagramas de casos de uso donde se refleja las funcionalidades desarrolladas, una para la parte web y otra para la parte móvil. A continuación, vemos un diagrama de casos de uso de la página web para mostrar las funcionalidades desarrolladas.

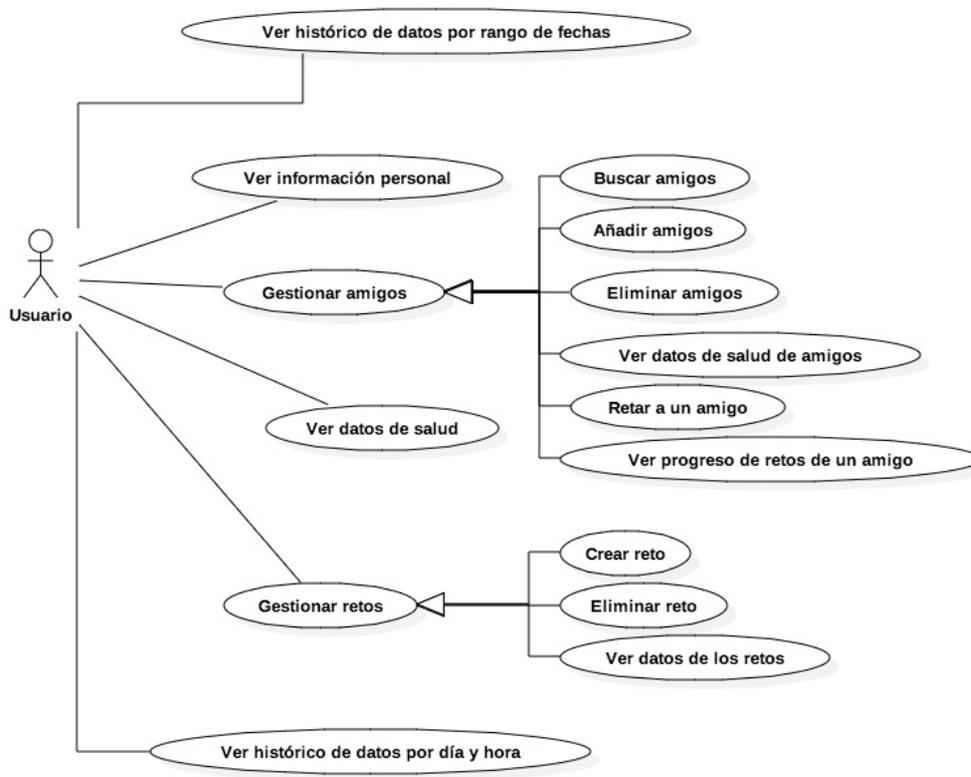


Figura 5.6: Figura que ilustra las funcionalidades de la página web con casos de uso

Como podemos apreciar se han desarrollado las funcionalidades básicas de la aplicación como ver los datos del usuario, las funcionalidades sociales, como los amigos y retos, así como, el histórico de los datos. En la siguiente figura ilustra las funcionalidades que tiene la aplicación móvil.

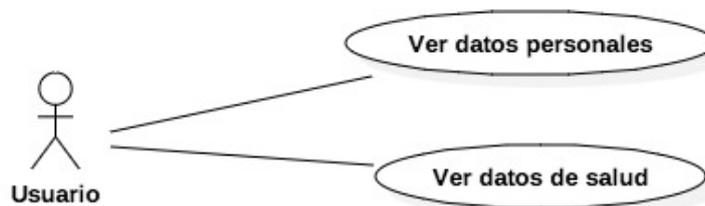


Figura 5.7: Figura que ilustra las funcionalidades de la aplicación móvil con casos de uso

5.2. Diseño

En esta etapa debido a la limitación temporal y de conocimientos se tuvo que elegir desarrollar una web con las funcionalidades y una aplicación móvil que sirviese como recolector de los datos de salud con menos funcionalidades.

5.2.1. Arquitectura

A continuación, se mostrará la arquitectura del sistema donde podremos ver el flujo de los datos desde el origen hasta la página web.

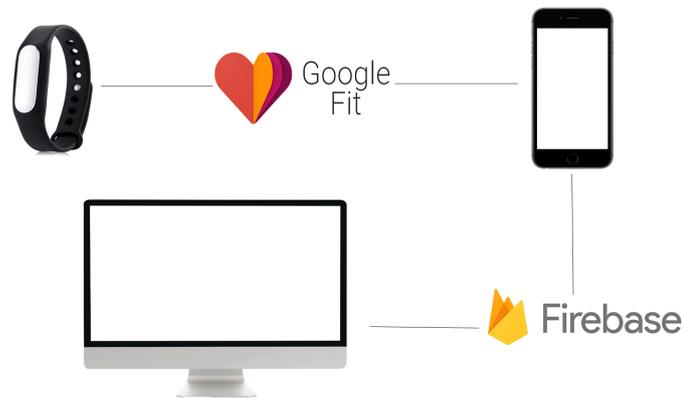


Figura 5.8: Figura que ilustra el flujo de información a través de los dispositivos y servicios

En esta imagen apreciamos que los datos se obtienen de la Xiaomi Mi Band 1S y estos se sincronizan con la aplicación Google Fit, nosotros utilizamos nuestra propia aplicación móvil para sincronizar los datos de Google Fit y almacenarlos en Firebase. Por último, la web se alimenta de los datos de Firebase.

5.2.2. Base de datos

A continuación se muestra una imagen de la estructura de la base de datos.

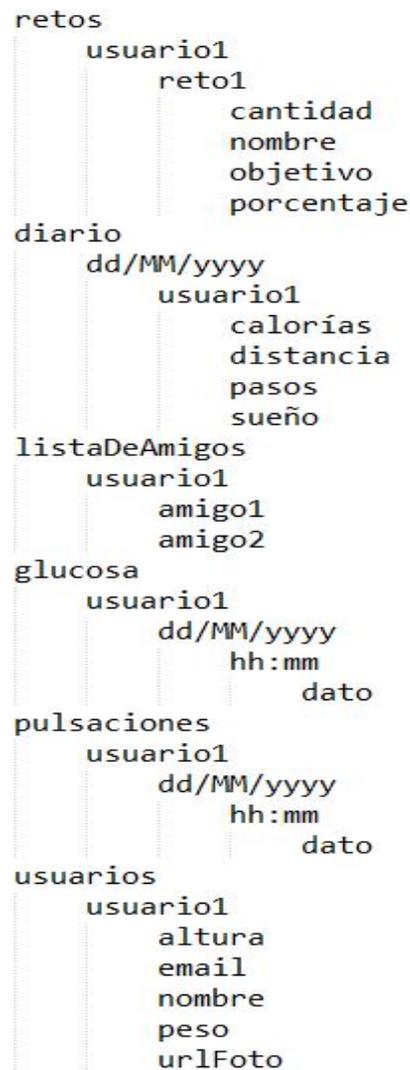


Figura 5.9: Figura que muestra la estructura de la base de datos

La base de datos de Firebase es no relacional y consta de un fichero json donde se escriben los datos estructurados de esa forma. Firebase nos recomienda tener el menos número de niveles de profundidad posibles para agilizar la descarga de información o incluso duplicarla.

5.2.3. Diseño de las interfaces de usuario

En este apartado se mostrarán algunos de los diseños de las interfaces gráficas a desarrollar tanto en la aplicación móvil como en la página web. A continuación, se muestra el diseño de la aplicación móvil.

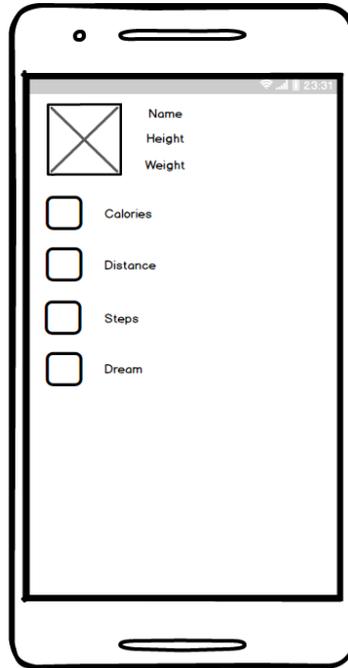


Figura 5.10: Figura que ilustra el diseño de la interfaz gráfica de la aplicación móvil

Como podemos ver el diseño es sencillo y muestra algunos de los datos básicos que guardamos de Google Fit y otros que a efectos prácticos se han tenido que introducir manualmente, debido a que Google Fit no nos proporciona todos los datos utilizados en este proyecto. La interfaz consta de un apartado de información del usuario junto a su imagen de perfil, así como la información de salud antes descrita. En la siguiente imagen vemos el diseño de la página principal de la web.

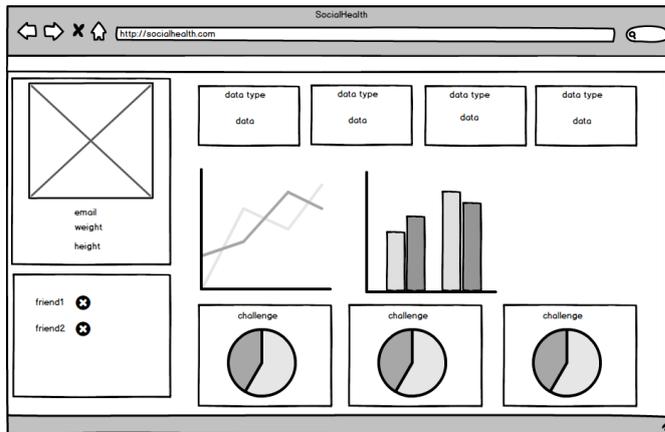


Figura 5.11: Figura que ilustra el diseño de la interfaz gráfica del perfil del usuario en la página web

En esta imagen observamos que al igual que la anterior aparecer una tarjeta con la información del usuario, otra tarjeta con la lista de amigos y al lado derecho aparecen los datos del usuario. Los datos diarios en la parte alta, los datos que se miden continuamente como pulsaciones o glucosa expresados en gráficas y por último los retos con su porcentaje de realización. La siguiente imagen muestra el diseño del apartado de amigos, donde se muestra los datos diarios de cada uno de ellos, así como, los retos de los mismos.

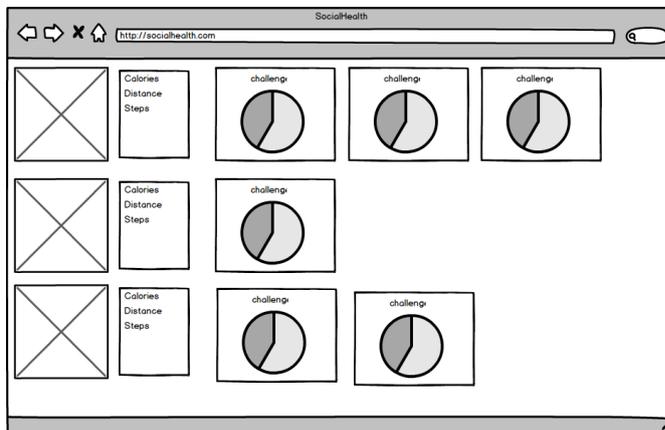


Figura 5.12: Figura que ilustra el diseño de la interfaz gráfica de la página de amigos en la página web

Podemos observar que esta página consta de una imagen de la información básica del usuario, así como sus datos diarios al lado derecho y los retos de cada uno de ellos.

5.2.4. Diagramas de clases

En esta sección se exponen los diagramas de clases tanto para la aplicación web como para la app móvil. A continuación, encontramos el diagrama de clases perteneciente a la aplicación móvil.

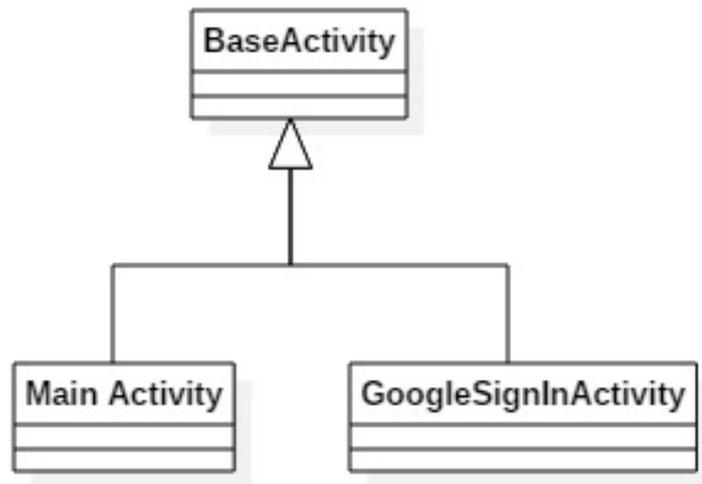


Figura 5.13: Figura que ilustra el diagrama de clases de la aplicación móvil

Como podemos observar nos encontramos con que se establece una clase llamada **BaseActivity** que nos sirve como base para las dos vistas que tiene la aplicación, la primera **GoogleSignInActivity** para el login y la segunda para mostrar los datos del usuario. La siguiente figura es el diagrama de clases de la página web.

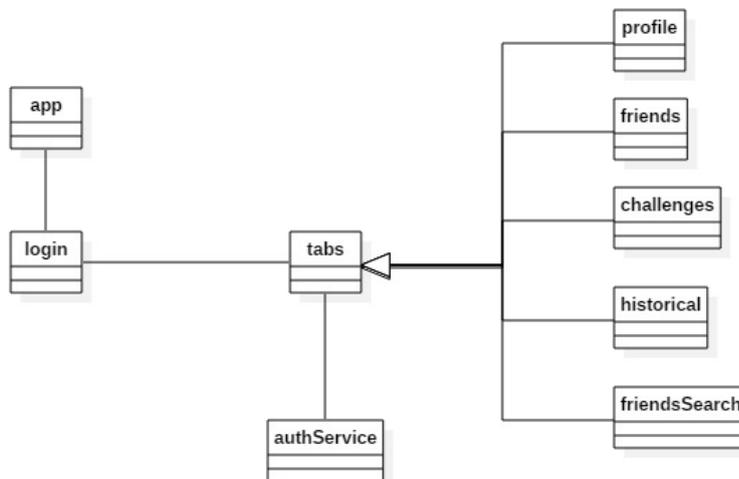


Figura 5.14: Figura que ilustra el diagrama de clases de la página web

En este diagrama observamos que la aplicación consta de una serie de pestañas o páginas diferentes. Estas páginas usan un servicio llamado authService que les permite hacer las peticiones de datos del usuario.

5.3. Implementación

Aunque, se ha mencionado en algunas partes del documento, hay datos que se usan en el proyecto que no se han obtenido por medio del hardware, sino que se han introducido. Para la aplicación móvil tanto la distancia como el sueño se introduce manualmente. Además, los datos de glucosa y pulsaciones también son generados, en este caso por un sencillo programa hecho en Java que genera un fichero con los datos en un rango determinado un número determinado de veces, por ejemplo, para treinta días se crean 43200 registros en el formato de json de Firebase.

A continuación, se exponen las tecnologías utilizadas para el desarrollo del proyecto.

5.3.1. Java

Es un lenguaje de programación de propósito general, concurrente, orientado a objetos, que ha sido diseñado para tener el menor número de dependencias posibles. La intención era que los desarrolladores escribieran una vez el programa y este pudiera ejecutarse en cualquier dispositivo, por lo

que este no tendría que ser recompilado para correr en otra máquina, esto lo consigue compilando en bytecode y ejecutando el código compilado en la máquina virtual de Java. En junio de 2017, Java es el lenguaje de programación más usado[41], sobre todo para aplicaciones cliente-servidor, este lenguaje fue desarrollado por James Gosling de Sun Microsystems y luego adquirida por Oracle.

Los objetivos principales por los que se creó Java son los siguientes:

1. Debería usar el paradigma de la programación orientada a objetos.
2. Debería permitir la ejecución de un mismo programa en múltiples sistemas operativos.
3. Debería incluir por defecto soporte para trabajo en red.
4. Debería diseñarse para ejecutar código en sistemas remotos de forma segura.
5. Debería ser fácil de usar y tomar lo mejor de otros lenguajes orientados a objetos, como C++.

Este lenguaje se utilizó para desarrollar la interfaz gráfica del sistema operativo Android y además es uno de los lenguajes soportados para la programación de aplicaciones para la plataforma. Android no tiene la máquina virtual Java, por lo que cuando se descarga una aplicación esta se compila en bytecode para el Android Runtime, una máquina diseñada específicamente para Android y optimizada para dispositivos móviles.

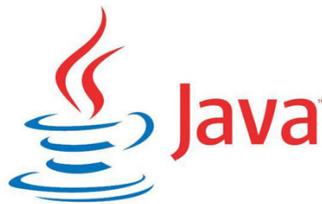


Figura 5.15: Logo de Java[36]

5.3.2. Android

Android es un sistema operativo basado en el núcleo Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos

inteligentes, tablets, relojes inteligentes, televisores y coches. Fue desarrollado por Android Inc, hasta que Google compró la empresa en 2005. Android es el sistema operativo móvil más usado en el mundo, esto se puede ver en *Kantar*, una página web donde refleja en porcentajes el número de dispositivos para cada sistema operativo móvil en función del país.



Figura 5.16: Logo de Android[21]

Sin embargo, la plataforma no ha estado exenta de problemas como, por ejemplo, la fragmentación que sufre la plataforma al no actualizar los dispositivos a las nuevas versiones por lo que estando en la séptima versión del sistema operativo vemos como entre la sexta(Mashmallow) y quinta versión(Lollipop) ocupan más del sesenta por ciento de los dispositivos Android como se puede apreciar en la siguiente imagen.

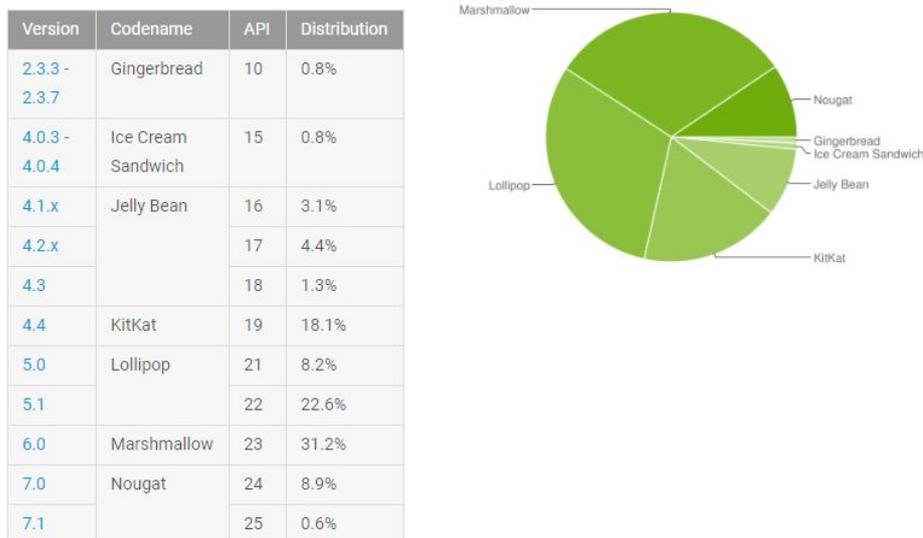


Figura 5.17: Imagen que ilustra la fragmentación de las versiones de Android[2]

5.3.3. TypeScript

Es un lenguaje de programación libre y de código abierto desarrollado y mantenido por Microsoft. Es un superconjunto de JavaScript, que esencialmente añade tipado estático y objetos basados en clases. Este lenguaje se puede usar para desarrollar aplicaciones grandes aplicaciones JavaScript, tanto para el lado del cliente como del servidor. Al ser un superconjunto de JavaScript, este puede ejecutar cualquier código en JS. Además, se puede usar cualquier librería apta para JavaScript. TypeScript tiene un compilador escrito a su vez en el mismo lenguaje que se transcompila a JavaScript. A continuación se muestra una imagen con una porción de código de muestra.

```
class Person {
  private name: string;
  private age: number;
  private salary: number;

  constructor(name: string, age: number, salary: number) {
    this.name = name;
    this.age = age;
    this.salary = salary;
  }

  toString(): string {
    return `${this.name} (${this.age}) (${this.salary})`;
  }
}
```

Figura 5.18: Código de muestra escrito en Typescript[2]

Como podemos observar en la figura, la sintaxis de TypeScript es muy parecida a lo que podríamos ver en Java por lo que lo convierte en un lenguaje fácil de aprender si vienes de Java o C++.



Figura 5.19: Logo de TypeScript[34]

5.3.4. Angular

Es un framework open-source para aplicaciones escrito en TypeScript y desarrollado por Google. Actualmente se encuentra en su cuarta versión y todas son compatibles excepto la primera versión AngularJS la cual fue escrita en JavaScript y posteriormente se rediseñó completamente con TypeScript. Este lenguaje se utiliza para crear y mantener aplicaciones web de una sola página, aunque también nos permite crear aplicaciones móviles nativas combinándolo con Ionic o React Native. Además, tiene como objetivo aumentar las aplicaciones basadas en navegador con el patrón Modelo Vista Controlador(MVC) la cual se puede ver a continuación en una imagen.

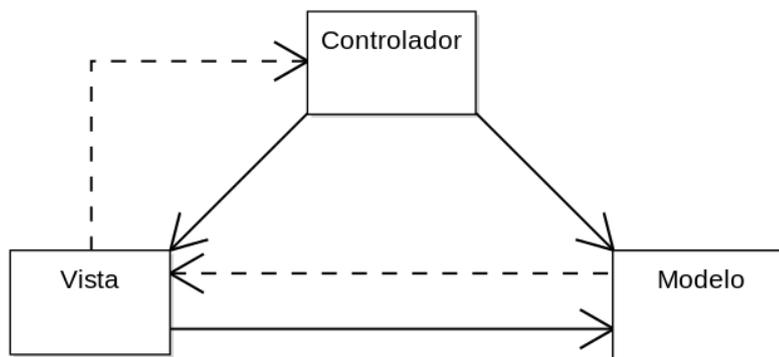


Figura 5.20: Figura que muestra el patrón arquitectónico Modelo Vista Controlador (MVC)[45]

A continuación, podemos ver una imagen que describe la arquitectura de Angular.

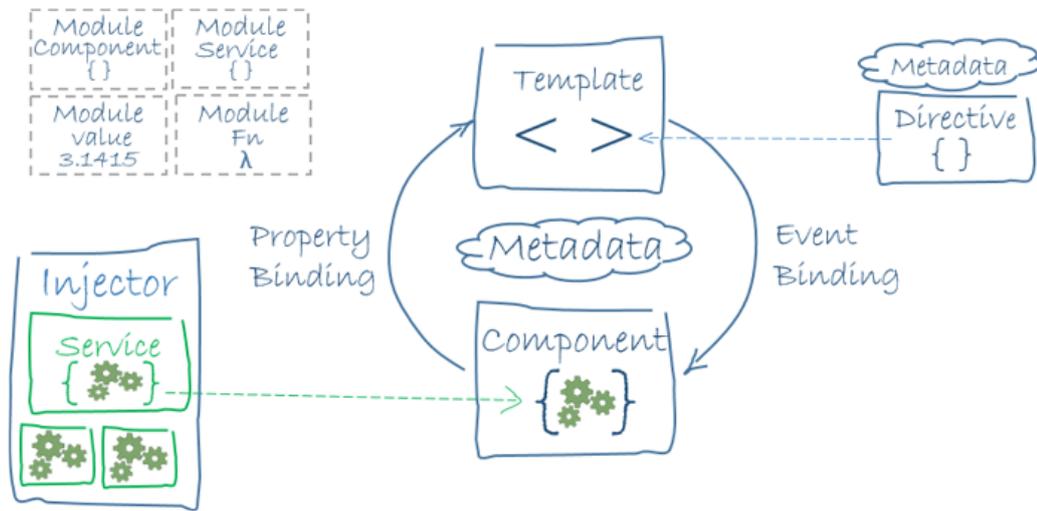


Figura 5.21: Figura que ilustra la arquitectura de Angular[4]

En esta imagen, podemos ver como los componentes de Angular manejan la plantilla en HTML mientras que los servicios componen la lógica de la aplicación. Además, los servicios y componentes se encapsulan en módulos.



Figura 5.22: Logo de Angular[23]

5.3.5. Ionic

Es un framework open-source tanto aplicaciones móviles como, para aplicaciones web progresivas. Este framework está hecho sobre Angular y Apache Cordova y nos permite usar las tecnologías web más usadas como HTML, CSS y JavaScript.

La finalidad de Ionic es mantener una apariencia de la interfaz del usuario atractiva y uniforme, así como las interacciones de la interfaz con la lógica, esto significa que no es un sustituto de JavaScript u otros frameworks, por lo que es perfecto para simplificar el desarrollo del front-end de una aplicación. También hay que señalar que una aplicación desarrollada con Ionic y Angular se puede convertir en una aplicación móvil tanto de Android como

de iOS, por lo que la convierte en una herramienta que nos permite desarrollar una sola vez y ejecutar en distintos dispositivos. En la siguiente imagen podemos apreciar la arquitectura de una aplicación nativa hecha con Ionic y Apache Cordova.

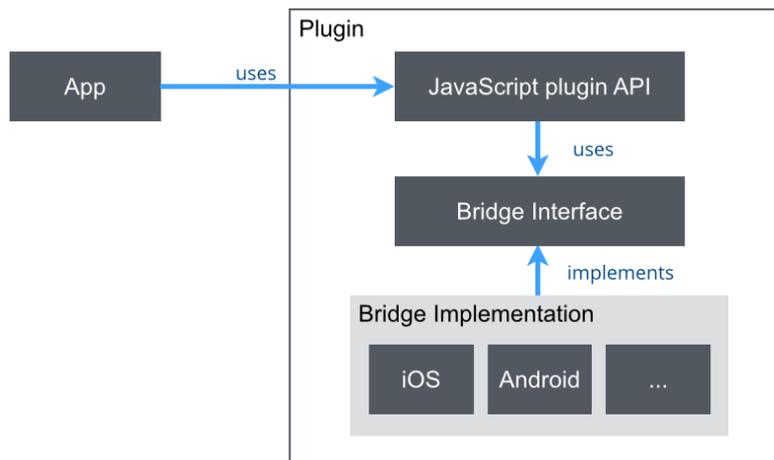


Figura 5.23: Arquitectura de Ionic con Apache Cordova[18]

Como se puede apreciar en la imagen las aplicaciones nativas se comunican con un plugin web por lo que no se traduce todo el código a otro lenguaje.



Figura 5.24: Logo de Ionic[19]

5.3.6. Firebase

Firebase es una plataforma de desarrollo para móviles y aplicaciones web desarrollada por Google. Se compone de diversas características que cada desarrollador puede usar en función de sus necesidades. A continuación, se muestran sus funcionalidades:

- Analítica.

- Mensajería en la Nube.
- Autenticación.
- Base de datos en tiempo real.
- Almacenamiento.
- Alojamiento web.
- Configuración remota.
- Laboratorio de pruebas.
- Notificaciones.
- Reporte de fallos.
- Indexación de aplicaciones.
- Invitaciones.
- Links Dinámicos

Además, Firebase ofrece una documentación extensiva sobre sus funcionalidades con ejemplos rápidos en varios lenguajes como Android, iOS, JavaScript, C++ o Unity. Además, hay que tener en cuenta que existen varias librerías de código libre para otros lenguajes como, en el caso de este proyecto, Angularfire2 una librería hecha para Angular y Firebase.

Para este TFG se han usado dos características, la primera es la autenticación, la cual nos permite iniciar sesión de las siguientes formas:

- Correo electrónico y contraseña.
- Teléfono.
- Google.
- Facebook.
- Twitter.
- GitHub.
- Anónimo.

De estos métodos de sesión se ha usado el de Google puesto que al necesitar de los datos de Google Fit se minimizaban los datos de las cuentas. La segunda funcionalidad es la base de datos en tiempo real. Esta base de datos no relacional se compone de un fichero json que se puedes descargar y subir para modificar la base de datos y que además nos ofrece la posibilidad de hacer copias de seguridad, introducir reglas de entrada para mayor seguridad y ver el uso de la base de datos en tiempo real.



Figura 5.25: Logo de Firebase[32]

5.3.7. HTML5

Es un lenguaje de marcado para la elaboración de páginas web desarrollado por World Wide Web Consortium. Este lenguaje tiene como filosofía de desarrollo la diferenciación, por lo que para añadir un elemento externo a la página, este no se incrusta, sino que se hace referencia a él con texto. De este modo recae en el navegador interpretar el código y visualizar la web. Como HTML es un estándar, este busca ser un lenguaje que permite que cualquier página web pueda ser interpretada de la misma forma en cualquier navegador actualizado.



Figura 5.26: Logo de HTML5[47]

5.3.8. CSS3

Es un lenguaje de diseño gráfico desarrollado por World Wide Web Consortium para definir la presentación de un documento escrito en un lenguaje de marcado como HTML. Este se utiliza para establecer el estilo de la interfaz gráfica de una página web, así como, puede ser usado en documentos XML, RSS o SVG. También hay que destacar que permite aplicar estilos no visuales con las hojas de estilo auditivas.



Figura 5.27: Logo de CSS[46]

5.3.9. Sass

Es un metalenguaje open-source de hojas de estilo en cascada (CSS) que además de componer de SassScript, el lenguaje de script que utiliza. Este, al igual que TypeScript, se traduce directamente a CSS. La mejora que aporta este lenguaje es que extiende las capacidades de CSS, incluyendo características de otros lenguajes de programación, como, por ejemplo, lenguajes orientados a objetos. Actualmente hay dos sintaxis diferentes del lenguaje una corresponde a SASS y la otra a SCSS. La primera se diferencia en que se usa la indentación para separar los bloques de código, así como líneas vacías para separar las reglas. A continuación, un ejemplo de la misma.

```
$blue: #3bbfce
$margin: 16px

.content-navigation
  border-color: $blue
  color: darken($blue, 9%)

.border
  padding: $margin/2
  margin: $margin/2
  border-color: $blue
```

Figura 5.28: Ejemplo código SASS[25]

Sin embargo, SCSS al parecerse a CSS tiene el siguiente aspecto para el mismo código.

```
$blue: #3bbfce;
$margin: 16px;

.content-navigation {
  border-color: $blue;
  color:
    darken($blue, 9%);
}

.border {
  padding: $margin / 2;
  margin: $margin / 2;
  border-color: $blue;
}
```

Figura 5.29: Ejemplo código SCSS[26]

Las características principales de Sass son las siguientes:

- Variables.
- Anidamiento.
- Mixins.
- Herencia de selectores.



Figura 5.30: Logo de Sass[27]

Capítulo 6

Conclusiones y trabajo futuro

En este capítulo se expondrán las conclusiones, así como, el trabajo futuro.

6.1. Conclusiones

En el desarrollo de este proyecto se ha realizado una aplicación Android que recolecta datos de Google Fit por lo que me ha permitido tener un acercamiento a la programación para Android, aunque haya sido poco. Además, se ha desarrollado una aplicación web con algunas de las tecnologías más usadas y punteras en las que se trabaja actualmente, permitiendo ampliar el abanico de lenguajes y tecnologías que conozco.

La idea de este proyecto surgió gracias a que en primer lugar disponía del hardware y, en segundo lugar, que deseaba hacer un TFG que pudiera repercutir en la sociedad de algún modo. Por lo tanto, ha sido muy gratificante en lo personal desarrollar software desde cero y sin ninguna experiencia previa en los lenguajes, por lo que ha resultado más difícil de lo esperado. Por otra parte, creo que el aspecto más importante y difícil del proyecto ha sido la planificación, debido en parte a los conocimientos de los lenguajes y en parte a que este es el primer proyecto que abarco solo.

Para concluir, quiero decir que este proyecto ha sido un verdadero reto, pero a pesar de eso se han conseguido cumplir los objetivos planteados al principio de este documento y que además ha servido para saber hacia dónde enfocar mi carrera profesional.

6.2. Trabajo futuro

En el caso de seguir desarrollando este proyecto se deberían tener en cuenta además de aquellas historias de usuario no hechas los siguientes puntos:

- Crear una aplicación universal. Para ello habría que modificar la web para que fuese la que se comunicara con Google Fit u otros servicios y se utilizara la tecnología de Ionic para convertir la web en una aplicación tanto para iOS como para Android y así poder eliminar la aplicación móvil en Java.
- Incrementar la cantidad de datos a mostrar como, por ejemplo, velocidad media en una actividad o el oxígeno en sangre.
- Incluir otros servicios de los que sincronizar datos o dispositivos específicos.
- Establecer publicidad agradable para el usuario para monetizar la aplicación.
- Ofrecer a investigadores datos de los usuarios que accedan a ello con el fin de recolectar los datos para las investigaciones.
- Realizar una aplicación para SmartWatch Android Wear, WatchOS y Tizen.

Apéndice A

Manual de usuario

En este capítulo se expondrá los resultados del desarrollo así como la información requerida para el uso de las aplicaciones.

A.1. Aplicación Smartphone

Cuando instalamos la aplicación y la iniciamos por primera vez nos aparecerá la siguiente pantalla.



Figura A.1: Pantalla de inicio de sesión de la aplicación móvil

En esta figura encontramos un botón de inicio de sesión con Google. Si solo tenemos una cuenta en el móvil iniciará sesión automáticamente con la primera si no, podrás elegir la cuenta. Además, si es la primera vez que inicias sesión la aplicación te preguntará si aceptas compartir tus datos personales. A continuación, se muestra una vez iniciada sesión la pantalla principal.

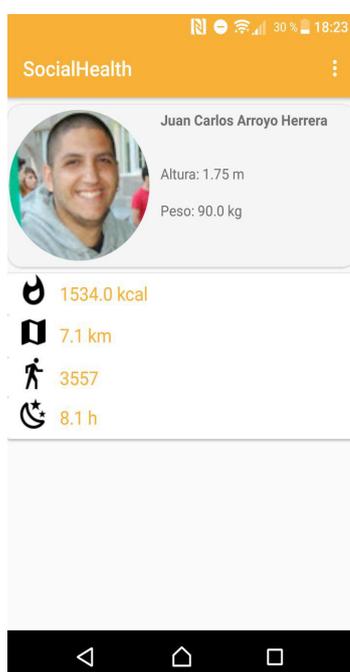


Figura A.2: Pantalla principal que muestra los datos del usuario

En esta figura podemos observar los datos básicos del usuario como la altura el peso o el nombre, así mismo, podemos los datos diarios obtenidos a través de Google Fit, excepto la distancia y el sueño que Google Fit no lo proporciona. Además, en la esquina superior derecha vemos el menú. En la siguiente imagen vemos que el menú desplegado consta de un botón para cerrar sesión.

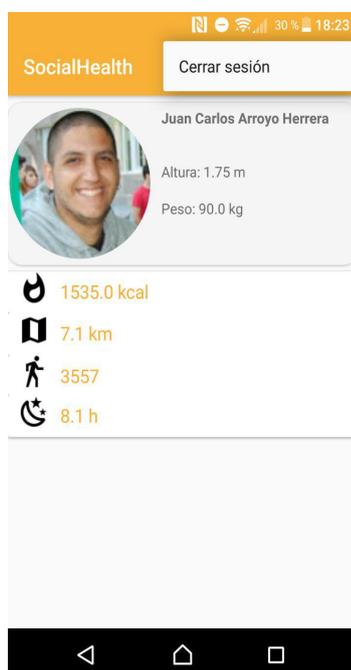


Figura A.3: Pantalla principal que muestra los datos del usuario y el botón de cerrar sesión

A.2. Aplicación web

A.2.1. Inicio de sesión

En la siguiente imagen vemos la pantalla de bienvenida de la página web, con el botón que nos permite iniciar sesión usando nuestra cuenta de Google.

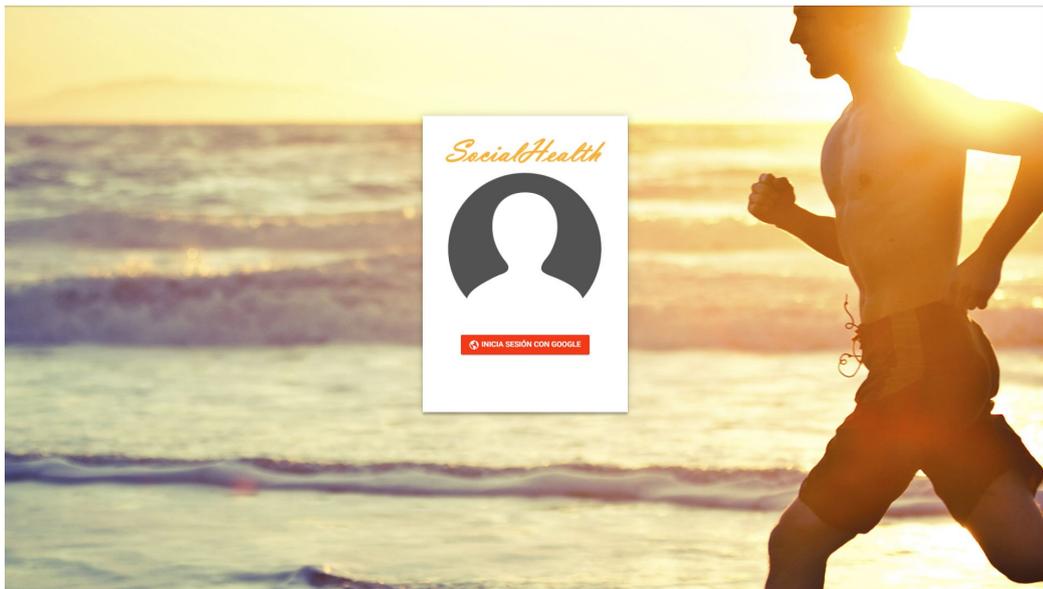


Figura A.4: Página principal de inicio de sesión de la página web

Google nos redirige a otra página donde seleccionamos la cuenta si esta ya existe en el navegador o introducimos una nueva.

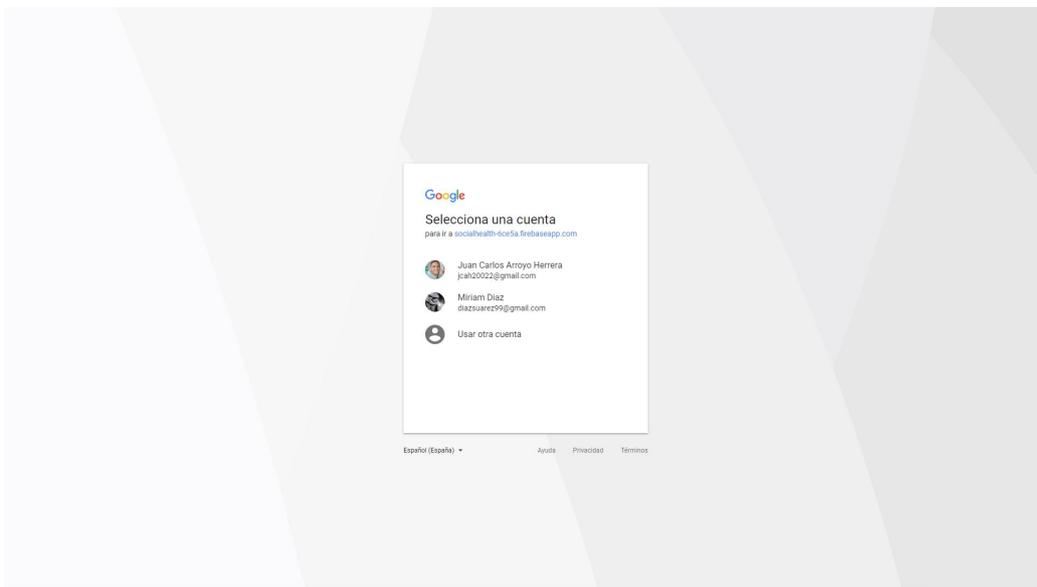


Figura A.5: Página de redireccionamiento del inicio de sesión de Google

A.2.2. Inicio

Una vez se ha iniciado sesión se muestra la vista principal de la aplicación. Pero primero veamos la estructura de la página web en la siguiente imagen.

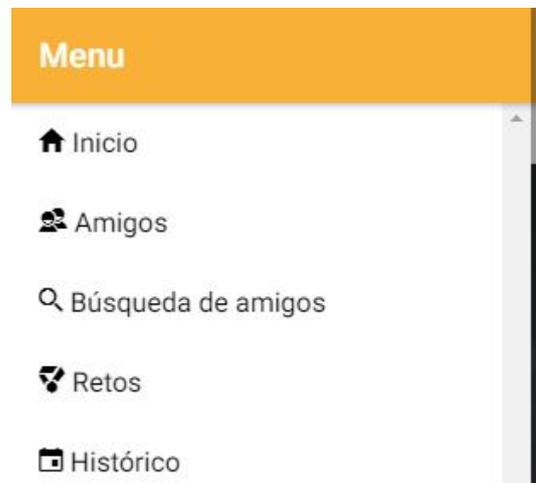


Figura A.6: Menú lateral de la página web

A continuación, se muestra la página inicial de la web. En ella podemos observar al igual que en la aplicación móvil, los datos de usuario. Debajo de la tarjeta del usuario encontramos la lista de amigos con el botón para

eliminarlos, a la derecha podemos ver una primera fila con los datos diarios del usuario, una segunda fila con los datos de la última hora de glucosa y pulsaciones en un gráfico.

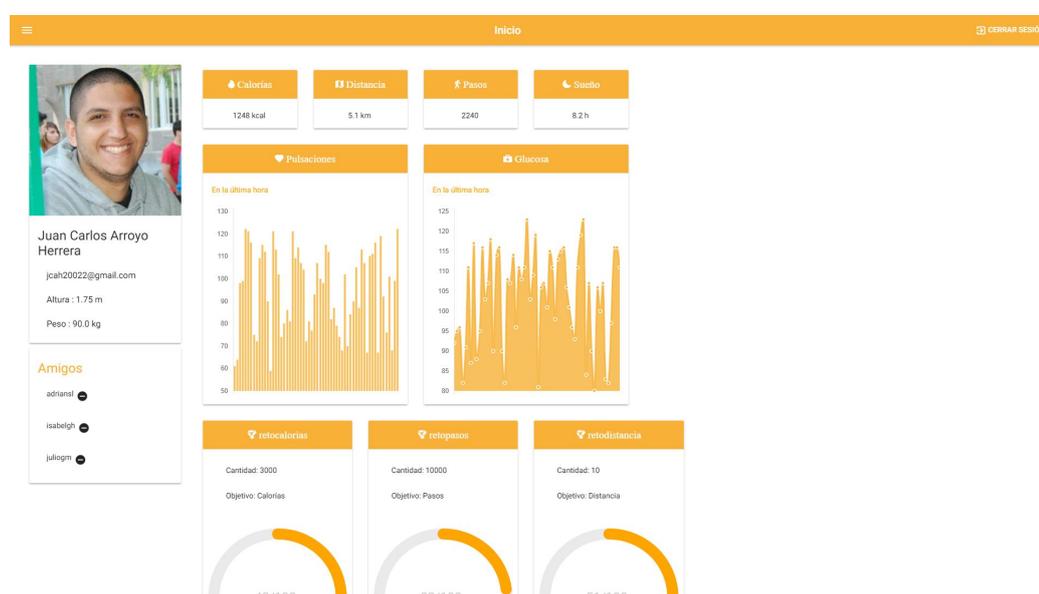


Figura A.7: Primera parte de la página que muestra el perfil personal del usuario con sus datos

En esta imagen de la página inicial vemos la última fila de los datos del usuario, en este caso los retos. Los retos se componen cada uno de una tarjeta con el nombre del reto, el tipo de dato y la cantidad, así como un gráfico que indica el porcentaje de progreso de cada reto.

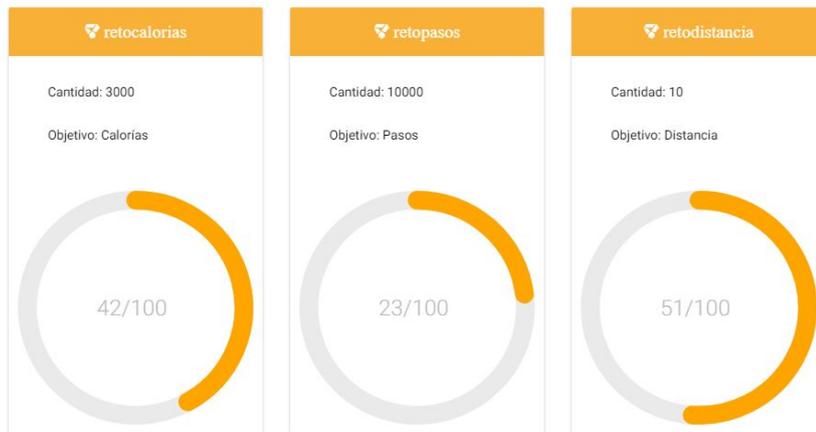


Figura A.8: Segunda parte de la página que muestra el perfil personal del usuario con sus datos

A.2.3. Amigos

A continuación, usando el menú lateral podemos ver la página de los amigos donde se muestra la información básica de cada amigo, su foto, nombre, así como sus datos diarios y los retos de cada uno de ellos.

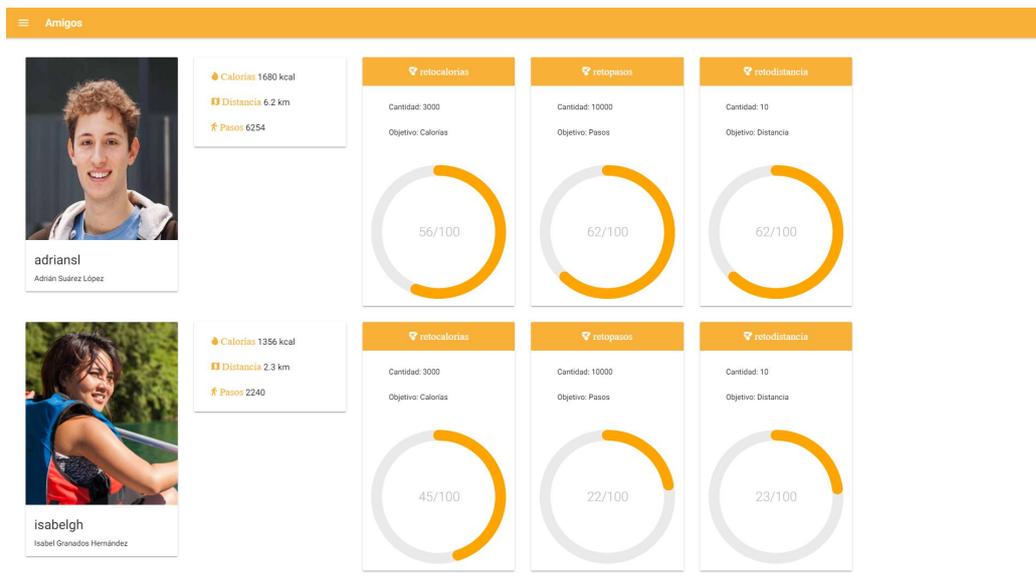


Figura A.9: Página que muestra a los amigos y sus datos y retos

A.2.4. Búsqueda de amigos

En esta página mientras escribe en el campo de búsqueda se irán filtrando los resultados los nombres de usuario y estos tendrán un botón se seguir para añadirlos a la lista de amigos. En la siguiente imagen veremos la página de los retos.



Figura A.10: Página que muestra el buscador de usuarios

A.2.5. Retos

Esta página se compone de un formulario con el que puedes crear los retos y además, retar a tus amigos. También, aparecen los retos del usuario, los cuales se pueden eliminar.

The screenshot shows the 'Retos' page with the following elements:

- Header:** A horizontal orange bar containing a menu icon (three horizontal lines) and the text 'Retos'.
- Form 'Añadir un reto':** A white box with an orange border containing:
 - Nombre:** A text input field with a small note: 'Puede usar letras, números, puntos y guión bajo.'
 - Calorías:** A radio button.
 - Pasos:** A radio button.
 - Distancia:** A radio button.
 - Cantidad:** A text input field.
 - Reta a tus amigos:** A section with two checkboxes: 'adriansl' and 'isabelgh'.
 - AÑADIR:** An orange button at the bottom.
- Challenge Cards:** Three white boxes with orange headers and 'ELIMINAR' buttons:
 - retocalorias:** Cantidad: 3000, Objetivo: Calorías.
 - retopasos:** Cantidad: 10000, Objetivo: Pasos.
 - retodistancia:** Cantidad: 10, Objetivo: Distancia.

Figura A.11: Página que muestra la sección de retos de la aplicación web

A.2.6. Histórico

Por fecha

En la siguiente figura observamos que tenemos un formulario donde seleccionas un día y la categoría todos, se mostrarán los datos diarios del día seleccionado.

The screenshot shows a mobile application interface for a 'Historico' (History) page. The page has an orange header with a hamburger menu icon and the title 'Historico'. Below the header, there are two main sections. On the left is a form for selecting a date and category. It has two tabs: 'DÍA' (selected) and 'INTERVALO'. Under the 'DÍA' tab, there is a text input field with the value '27-05-2017', a red 'X' icon, and a calendar icon. Below this is a section for selecting a type, with three radio buttons: 'Pulsaciones', 'Glucosa', and 'Todos' (which is selected). There is also a section for entering a time, with a text input field showing '22:56'. At the bottom of the form is an orange 'ACEPTAR' button. On the right side of the page, the date '27-05-2017' is displayed. Below this, there are two summary cards: 'Calorías' (2406.0 kcal) and 'Distancia' (4.8 km).

Figura A.12: Página que muestra el histórico de todos los datos para un día determinado

En la siguiente imagen vemos que al seleccionar una fecha, pulsaciones o glucosa y una hora determinada, se nos mostrará en una gráfica los datos de esa hora en esa fecha.

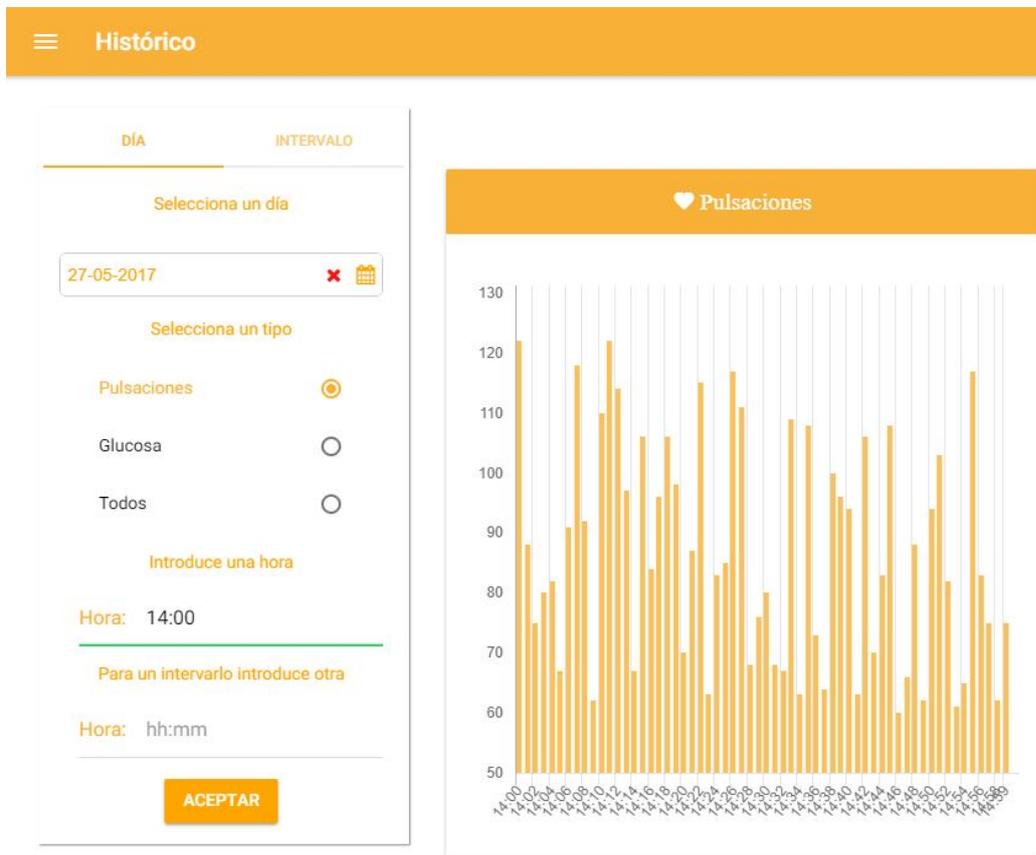


Figura A.13: Página que muestra el histórico de pulsaciones para una hora determinada de un día

En la siguiente figura se muestra que si deseamos un rango de horas de un día determinado para pulsaciones o glucosa, se nos mostrará una gráfica con los datos.

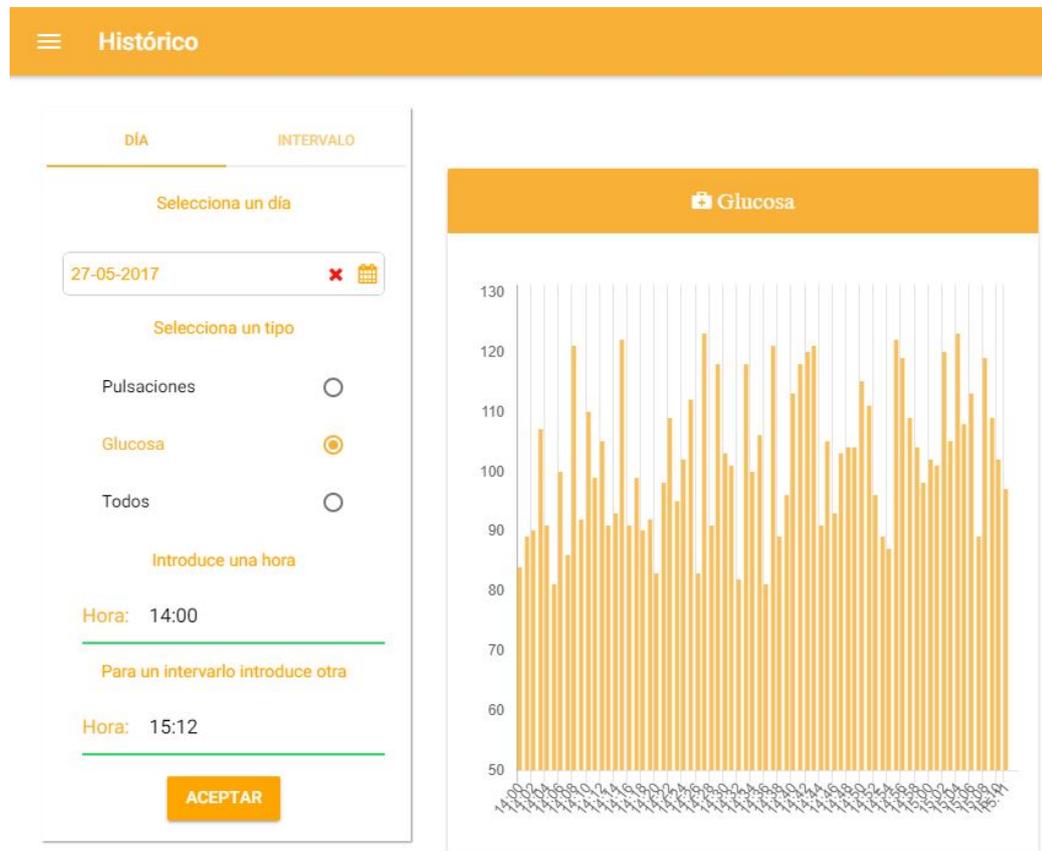


Figura A.14: Página que muestra el histórico de glucosa para un rango determinado de horas de un día

A.2.7. Por intervalo de fechas

Si queremos ver todos los datos para un rango de fechas debemos seleccionar el rango y la categoría todos. A continuación se mostrarán las gráficas para cada tipo de dato diario, como se puede ver en la siguiente imagen.

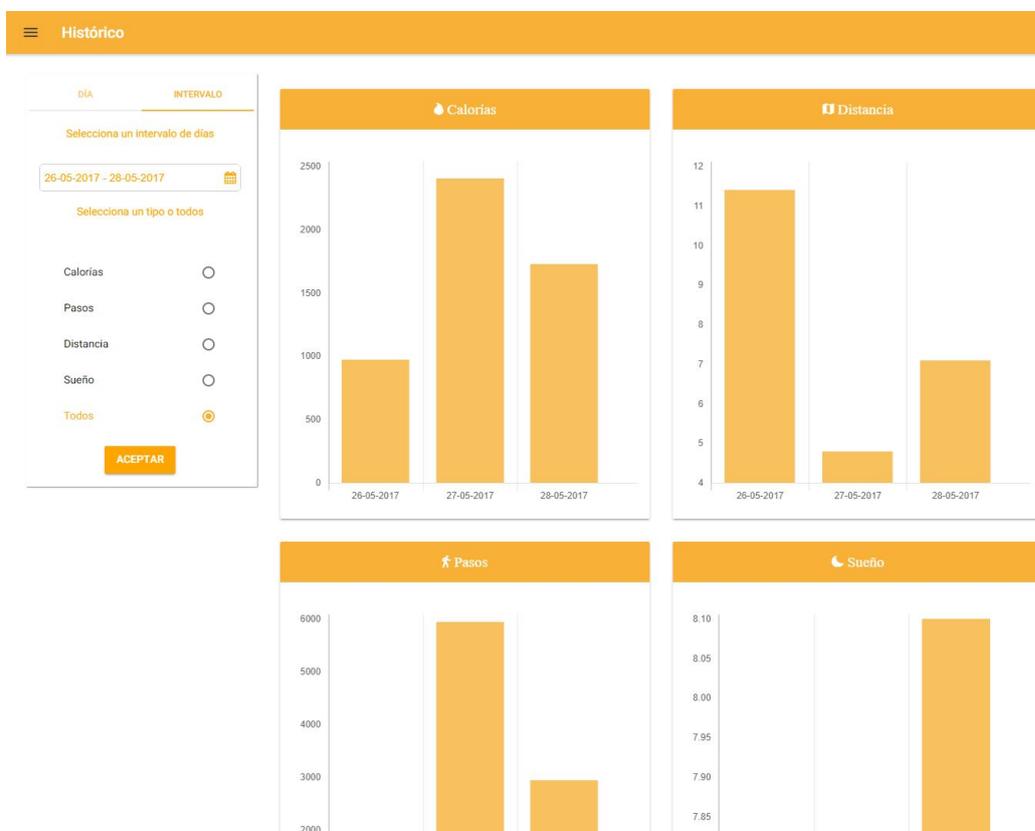


Figura A.15: Página que muestra gráficas de todos los datos diarios en un rango determinado de fechas

Si por otro lado deseamos ver un solo tipo de dato diario, bastará con seleccionar la categoría elegida como se observa en la siguiente figura.

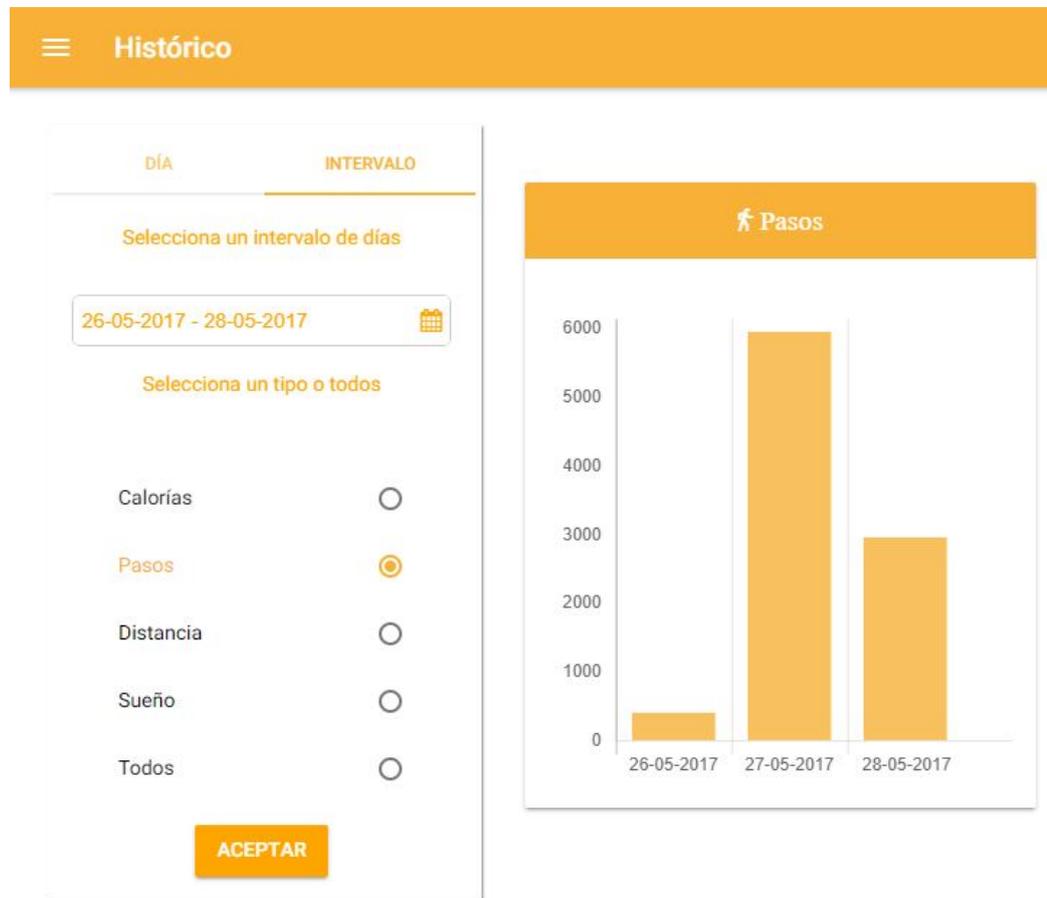


Figura A.16: Página que muestra una gráfica del tipo de dato diario en un rango determinado de fechas

Apéndice B

Competencias específicas cubiertas

La elaboración de este proyecto ha llevado a cubrir una serie de competencias específicas de la mención Ingeniería de Software. A continuación, se presentan las competencias cubiertas en este TFG:

- **IS01. Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos y prácticas de la ingeniería del software:**

La competencia expuesta queda cubierta con la realización del proyecto. En el capítulo cinco se ve con mayor detalle el cumplimiento de la competencia.

- **IS02. Capacidad para valorar las necesidades del cliente y especificar los requisitos software para satisfacer estas necesidades, reconciliando objetivos en conflicto mediante la búsqueda de compromisos aceptables dentro de las limitaciones derivadas del coste, del tiempo, de la existencia de sistemas ya desarrollados y de las propias organizaciones:**

Esta competencia se ve cubierta en los capítulos cuatro y cinco, también debido a la utilización de Scrum como metodología.

- **IS03. Capacidad de dar solución a problemas de integración en función de las estrategias, estándares y tecnologías disponibles:**

La integración de la página web y la aplicación Android mediante Firebase, así como, la elección de patrones, lenguajes y herramientas, cubre esta competencia.

- **IS04. Capacidad de identificar y analizar problemas y diseñar, desarrollar, implementar, verificar y documentar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales:**

La propia elaboración de este proyecto conlleva el cumplimiento de esta competencia, en especial en los capítulos cuatro y cinco.

- **IS05. Capacidad de identificar, evaluar y gestionar los riesgos potenciales asociados que pudieran presentarse:**

Al igual que la primera esta competencia se ve cubierta por la utilización de Scrum en el proyecto.

Apéndice C

Normativa y Legislación

C.1. Ley de Protección de Datos

En este proyecto de final de grado se deben tener en cuenta una serie de requisitos legales debido a que se hace uso de datos de carácter personal, así como, de datos de salud. Por lo tanto, el sistema deberá cumplir dos leyes. La primera cumplir con el Reglamento de Seguridad del Real Decreto 994/1999 de la Ley Orgánica de Protección de Datos Personales (LOPD). La Ley Orgánica de Protección de Datos Personales tiene por objetivo garantizar y proteger, en lo que concierne al tratamiento de los datos personales, las libertades públicas y los derechos fundamentales de las personas físicas, y especialmente de su honor e intimidad personal y familiar. Esta Ley Orgánica será aplicada a los datos de carácter personal registrados en soporte físico, que los haga susceptibles de tratamientos, y a toda modalidad de uso posterior de estos datos por los sectores público y privado.

La segunda, que concierne al ámbito de la sanidad debe cumplir con la Ley General de Sanidad de 25 de abril de 1986 y la Ley 41/ 2002, de 14 de noviembre, básica reguladora de la Autonomía del Paciente y de derechos y obligaciones en materia de información y documentación clínica.

Estas leyes nos obligan a desarrollar medidas de seguridad para el correcto uso y la protección de los datos de salud del usuario. Por esto, se le pide al usuario que acepte expresamente que permita a la aplicación acceder y utilizar los datos personales y de salud.

Bibliografía

- [1] Imagen: ejemplo de desarrollo incremental. <http://docplayer.es/docs-images/22/1449660/images/30-0.png>". Accedido: 09-06-2017.
- [2] Google. Android. Imagen: figura que muestra la fragmentación de android. "https://developer.android.com/about/dashboards/index.html". Accedido: 09-06-2017.
- [3] Google. Angular. Documentación online de Angular. "https://angular.io/docs/ts/latest/". Inglés. Accedido: 09-06-2017.
- [4] Google. Angular. Imagen: arquitectura de una aplicación en angular. "https://angular.io/guide/architecture". Accedido: 09-06-2017.
- [5] Google. Angular. Repositorio público de Github de AngularFire2. "https://github.com/angular/angularfire2". Inglés. Accedido: 09-06-2017.
- [6] Google. Angular. Tutorial online "Tour of Heroes". "https://angular.io/docs/ts/latest/tutorial/". Inglés. Accedido: 09-06-2017.
- [7] Apple. Imagen: figura que muestra tres apple watch. "https://topesdegama.com/app/uploads/2016/12/apple-watch-apple-1024x585.jpg". Accedido: 09-06-2017.
- [8] Apple. Imagen: logo de la aplicación de iOS, Health. "https://www.sketchappsources.com/resources/source-image/AppleHealth.png". Accedido: 09-06-2017.
- [9] Apple. Imagen: logo del framework para cuidados personales CareKit. "https://avatars3.githubusercontent.com/u/18153292?v=3&s=200". Accedido: 09-06-2017.

- [10] Apple. Imagen: logo del framework para investigación ResearchKit. ”<http://www.cranstonit.com/blog/files/researchkit-rcm992x0.jpg>”. Accedido: 09-06-2017.
- [11] Balsamiq. Imagen: logo de Balsamiq Mockups. ”<http://deshack.net/wp-content/uploads/2015/07/balsamiq-big.jpg>”. Accedido: 09-06-2017.
- [12] Ryan Lienhart Dahl. Imagen: logo de Node.js. ”<https://pixabay.com/es/nodo-js-logo-nodejs-javascript-736399/>”. Accedido: 09-06-2017.
- [13] Equipo de Google +. Google. Documentación de Google Sign In. ”<https://developers.google.com/+/web/signin>”. Inglés. Accedido: 09-06-2017.
- [14] Equipo de Google Fit. Google. Documentación de Google Fit. ”<https://developers.google.com/fit/overview>”. Inglés. Accedido: 09-06-2017.
- [15] Equipo de Google Fit. Google. Imagen: logo de google fit. ”<https://exist.io/static/img/pics/logo-googlefit.png>”. Accedido: 09-06-2017.
- [16] Equipo de Ionic. Ionic. Documentación online de Ionic. ”<https://ionicframework.com/docs/>”. Inglés. Accedido: 09-06-2017.
- [17] Equipo de Valor Software. Valor Software. Repositorio público de Github de ng2-charts. ”<https://github.com/valor-software/ng2-charts>”. Inglés. Accedido: 09-06-2017.
- [18] Drifty. Imagen: figura que muestra la arquitectura de Ionic. ”<https://blog.codecentric.de/files/2014/11/architecture.png>”. Accedido: 09-06-2017.
- [19] Drifty. Imagen: logo de Ionic. ”https://1.bp.blogspot.com/-VxIXLOBv2ns/V1yoUEerOCI/AAAAAAAAAHU/VkPV8LB2C44-wqg5GLOE8kruf_9hh-GgCLcB/s1600/ionic.png”. Accedido: 09-06-2017.
- [20] Google. Firebase. Imagen: comparación de los diferentes planes que ofrece firebase. ”<https://firebase.google.com/pricing/>”. Accedido: 09-06-2017.
- [21] Google. Imagen: logo de Android. ”http://www.brandemia.org/wp-content/uploads/2012/10/logo_principal.jpg”. Accedido: 09-06-2017.

- [22] Firebase. Google. Documentación online de Firebase. "https://firebase.google.com/docs/". Inglés. Accedido: 09-06-2017.
- [23] Google. Equipo de Angular. Imagen: logo del framework Angular. "https://www.vectorlogo.zone/logos/angular/angular-card.png". Accedido: 09-06-2017.
- [24] Google. JetBrains. Equipo de Android Studio. Imagen: logo de Android Studio. "https://3.bp.blogspot.com/-BVQ36vhFc0I/VsOpnJmDI/AAAAAAAAAFk/Z4BOOKmbxJ4/s1600/banner.PNG". Accedido: 09-06-2017.
- [25] Hampton Catlin. Imagen: ejemplo de código SASS. "https://es.wikipedia.org/wiki/Sass_Accedido: 09-06-2017.
- [26] Hampton Catlin. Imagen: ejemplo de código SCSS. "https://es.wikipedia.org/wiki/Sass_Accedido: 09-06-2017.
- [27] Hampton Catlin. Imagen: logo de Sass (Syntactically Awesome Style Sheets). "http://sass-lang.com/assets/img/styleguide/color-1c4aab2b.png". Accedido: 09-06-2017.
- [28] Ramón Invarato. Libro online Android 100 %, Octubre 2014. "http://jarroba.com/libro-android-100-gratis/". Español. Accedido: 09-06-2017.
- [29] Kekeh. Repositorio público de Github de mydatepicker. "https://github.com/kekeh/mydatepicker". Inglés. Accedido: 09-06-2017.
- [30] Kekeh. Repositorio público de Github de mydaterangepicker. "https://github.com/kekeh/mydaterangepicker". Inglés. Accedido: 09-06-2017.
- [31] Kristiyan Kostadinov. Repositorio público de Github de angular-svg-round-progressbar. "https://github.com/crisbeto/angular-svg-round-progressbar". Inglés. Accedido: 09-06-2017.
- [32] Andrew Lee and James Tamplin. Imagen: logo de Firebase. "https://cdn.periscopix.co.uk/blog/_800xAUTO_crop_center-center_80/Firebase-Logo.png". Accedido: 09-06-2017.
- [33] Ash Maurya. *Running Lean: Iterate from Plan A to a Plan That Works*. O'Reilly Media, 2nd edition, 3 2012.

- [34] Microsoft Corporation. Imagen: logo de TypeScript. ”<http://15441-presscdn-0-77.pagely.netdna-cdn.com/wp-content/uploads/2015/05/TypeScript-logo.jpg>”. Accedido: 09-06-2017.
- [35] MKlab. Imagen: logo de SarUML. ”https://upload.wikimedia.org/wikipedia/en/7/71/StarUML_logo.png”. Accedido: 09-06-2017.
- [36] Oracle Corporation. Imagen: logo de Java. ”<http://fontslogo.com/wp-content/uploads/2013/03/Java-Logo-Font.jpg>”. Accedido: 09-06-2017.
- [37] Eric Ries. *The Lean Startup: How Constant Innovation Creates Radically Successful Businesses*. Portfolio Penguin, trade paperback edition, 10 2011.
- [38] Eduardo Miguel Rodríguez Barrera. Imagen: ciclo de experimentación en running lean. Apuntes de la asignatura Gestión del Software II. Running Lean -3- Identifica riesgos. Accedido: 09-06-2017.
- [39] Samsung. Imagen: logo de la aplicación de samsung health. ”https://lh3.googleusercontent.com/ohV24150stfzc8BiJrUEakjpy8-IDzAJaFV5YNhlpua0QV78A9LU_xI0AOpeHdVbZw=w300”. Accedido: 09-06-2017.
- [40] Jon Skinner. Imagen: logo de Sublime Text 3. ”<http://static.grayghostvisuals.com/imgblog/st2-logo.png>”. Accedido: 09-06-2017.
- [41] TIOBE Software BV. TIOBE Programming Community Index. ”<https://www.tiobe.com/tiobe-index/>”. Inglés. Accedido: 01-07-2017.
- [42] Unión Europea. eHealth. ”<https://ec.europa.eu/digital-single-market/en/policies/ehealth>”. Inglés. Accedido: 06-07-2017.
- [43] Benito van der Zander. Imagen: logo de TexStudio. ”<http://www.texstudio.org/images/texstudio128x128.png>”. Accedido: 09-06-2017.
- [44] PK Vitality. Imagen: figura que muestra el smartwatch para glucosa k’watch glucose. ”<http://www.pkvitality.com/wp-content/uploads/2016/12/glucosepng2.png>”. Accedido: 09-06-2017.
- [45] Wikipedia. Imagen: figura que muestra el patrón arquitectónico modelo vista controlador. ”<https://es.wikipedia.org/wiki/Modelo>”. Accedido: 09-06-2017.

- [46] World Wide Web Consortium. Imagen: logo de CSS. "http://heyba.by/wp-content/uploads/2014/01/CSS3-badge-logo-big.png". Accedido: 09-06-2017.
- [47] World Wide Web Consortium. Imagen: logo de HTML5. "https://www.w3.org/html/logo/downloads/HTML5_Logo_512.png". Accedido: 09-06-2017.
- [48] Xiaomi Inc. Imagen: SmartBand Xiaomi Mi Band 1S. "http://esmimovil.com/144-large_default/xiaomi-mi-band.jpg". Accedido: 09-06-2017.