



UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



TFG del Grado en Ingeniería  
Informática  
RedELA Multiplataforma



Presentado por Javier Martín Castro  
en Universidad de Burgos — 7 de julio de 2024  
Tutor: Pedro Renedo Fernández







UNIVERSIDAD DE BURGOS  
ESCUELA POLITÉCNICA SUPERIOR  
Grado en Ingeniería Informática



D. Pedro Renedo Fernández, profesor del departamento de Ingeniería Informática, área de Lenguajes y Sistemas Informáticos.

Expone:

Que el alumno D. Javier Martín Castro, con DNI 24392279C, ha realizado el Trabajo final de Grado en Ingeniería Informática titulado RedELA Multiplataforma de TFG.

Y que dicho trabajo ha sido realizado por el alumno bajo la dirección del que suscribe, en virtud de lo cual se autoriza su presentación y defensa.

En Burgos, 7 de julio de 2024

Vº. Bº. del Tutor:

D. Pedro Renedo Fernández





## Resumen

La idea principal de realizar este proyecto nace de poder facilitar la vida a personas con esclerosis [13] siendo una enfermedad autoinmune que se caracteriza por causar deterioro en el sistema nervioso. Esta enfermedad se puede manifestar como **esclerosis múltiple (EM)** [15] o **esclerosis lateral amiotrófica (ELA)** [17].

Este tipo de pacientes ya sea con **EM** o **ELA**, necesita una serie de atenciones y cuidados para que su día a día mejore dentro de la gravedad de la enfermedad y nuestra idea es crear una aplicación para gestionar las citas con profesionales de la salud. Con todo eso y finalmente para el desarrollo de la aplicación se cuenta con el asesoramiento de la **Asociación de ELA de Castilla y León (ELACyL)**. Los objetivos de esta asociación es dar visibilidad a aquellas personas afectadas para que la sociedad tome conciencia de la gravedad de la enfermedad, obtener más recursos y financiación para realizar investigaciones de la ELA y finalmente atender a los afectados por la enfermedad ya que necesitan una cantidad elevada de recursos para poder llevar una vida digna.

Para ayudar a estas personas se va a proceder a desarrollar una aplicación multiplataforma para la web y para dispositivos Android, en futuras versiones podrá desplegarse en dispositivos móviles Apple. Esta aplicación contará con un gestor de citas para que las personas puedan tener de manera amigable y cómoda todas las citas agendadas por el Gestor de casos.

## Descriptores

Esclerósis, Esclerosis Múltiple, Esclerosis Lateral Amiotrófica, ELA, ayuda, aplicación móvil Android, aplicación web, citas, pacientes, cuidadores principales.

## Abstract

The main idea behind this project is to facilitate the lives of individuals with sclerosis, an autoimmune disease that causes damage to the nervous system. This condition can manifest as **Multiple Sclerosis (MS)** [15] or **Amyotrophic Lateral Sclerosis (ALS)** [17].

Patients with **MS** or **ALS** require a range of care and attention to improve their daily lives within the severity of the disease. Our idea is to develop an application to manage appointments with healthcare professionals. With this goal in mind, we will be advised by the **ALS Association of Castilla and León (ELACyL)**. The objectives of this association are to raise awareness among those affected, to obtain more resources and funding for ALS research, and to provide support to those affected by the disease, as they require a significant amount of resources to live a dignified life.

To help these individuals, we will proceed to develop a multi-platform application for the web and Android devices, with future versions planned for deployment on Apple devices. This application will feature an appointment manager that allows individuals to easily and comfortably manage all scheduled appointments through a user-friendly interface.

Note: I translated "Asociación de ELA de Castilla y León" as "ALS Association of Castile and León", assuming that ELA stands for Amyotrophic Lateral Sclerosis, as mentioned earlier. If that's not the case, please let me know and I'll be happy to adjust the translation accordingly.

## Keywords

Sclerosis, Multiple Sclerosis, Amyotrophic Lateral Sclerosis, ALS, help, Android mobile app, web app, dating, patients, primary caregivers.



---

# Índice general

---

Índice general	iii
Índice de figuras	v
Índice de tablas	vi
<b>1. Introducción</b>	<b>1</b>
<b>2. Objetivos del proyecto</b>	<b>5</b>
2.1. Objetivos técnicos . . . . .	5
2.2. Objetivos personales . . . . .	6
<b>3. Conceptos teóricos</b>	<b>7</b>
3.1. Herramientas proyecto . . . . .	7
<b>4. Técnicas y herramientas</b>	<b>9</b>
4.1. Metodologías . . . . .	9
4.2. Patrón de diseño . . . . .	10
4.3. Repositorio . . . . .	11
4.4. Control de versiones . . . . .	11
4.5. Gestión del proyecto . . . . .	11
4.6. Entorno de desarrollo integrado (IDE) . . . . .	12
4.7. Integración y entregas continuas (CI/CD) . . . . .	14
4.8. Herramientas contrucción del proyecto . . . . .	14
<b>5. Aspectos relevantes del desarrollo del proyecto</b>	<b>17</b>
5.1. Inicio del proyecto . . . . .	17
5.2. Metodologías . . . . .	17
5.3. Formación . . . . .	18
5.4. Desarrollo . . . . .	19
5.5. Publicación . . . . .	19
<b>6. Trabajos relacionados</b>	<b>21</b>
<b>7. Conclusiones y Líneas de trabajo futuras</b>	<b>23</b>
7.1. Conclusiones . . . . .	23

7.2. Líneas de trabajo futuras . . . . .	23
<b>Bibliografía</b>	<b>25</b>

---

## Índice de figuras

---

4.1. Scrum . . . . .	9
4.2. Gitflow . . . . .	10
4.3. Arquitectura limpia . . . . .	11
4.4. Flutter . . . . .	12
4.5. Firebase . . . . .	13

---

# Índice de tablas

---

3.1. Herramientas y tecnologías utilizadas en cada parte del proyecto . . . . .	7
---	---

---

# 1. Introducción

---

Inicialmente el proyecto nacía para dar solución y mejorar la calidad de vida de personas que tengan **Esclerosis Múltiple** [15, 9] (**EM**). Para ello se iba a desarrollar el Frontend inicialmente para dispositivo móvil Android.

Resumiendo, podemos decir que la EM es una enfermedad neurológica crónica cuya naturaleza es inflamatoria y autoinmunitaria. Se caracteriza por la afectar el cerebro y la médula espinal y hace más lentos o bloquea la conexión entre el cerebro y el cuerpo. Algunos síntomas de EM pueden ser:

- Alteraciones en la visión.
- Debilidad a nivel muscular.
- Problemas con la coordinación y el equilibrio.
- Aparición de lesiones cerebrales detectadas en la resonancia magnética.

Podemos clasificar esta enfermedad entre:

- **Recurrente-remitente (EMRR)** caracterizada por aparición periódica de ataques o brotes.
- **Secundaria progresiva (EMSP)** se caracteriza por el empeoramiento gradual y progresiva de la discapacidad.
- **Primaria progresiva (EMPP)** este tipo comienza de manera inofensiva y empeora con el tiempo.
- **Progresiva-recurrente (EMPR)** este es el más agresivo ya que las apariciones de los síntomas son severas y periódicas.

Ante los problemas de diseño para poder desarrollar la aplicación y con la aprobación del tutor, Pedro Renedo Fernández, nos ponemos en contacto con al asociación de ELA de Castilla y León y realizamos una primera toma de contacto con ellos para saber que objetivos iniciales les parecería interesante para crear una aplicación multiplataforma que pudiera ayudar a pacientes o personas afectadas con la enfermedad. Al mismo tiempo, investigamos un poco de que trata la enfermedad y podemos afirmar que, la ELA que es una enfermedad degenerativa que daña el

sistema nervioso periférico y central. Va degenerando de manera progresiva las células motoras del cerebro y la médula espinal.

Tras analizar los objetivos marcados por la asociación y ver que la aplicación contará inicialmente con tres actores que serán **gestor de casos**, **paciente** y **cuidador principal**. Para tratar de desarrollar la aplicación nos centramos en dar solución a unos requisitos iniciales que les pareció interesante.

En una primera fase vamos y para dar comienzo al proyecto se va a realizar el desarrollo e implementación de los siguientes puntos:

- **Administrador.** Este rol tiene la posibilidad de realizar las siguientes gestiones:

- **Gestión usuarios**

- **Gestión administradores.** Esta gestión se define por poder gestionar aquellos usuarios que son **administradores** de la aplicación. En ella se podrán ver la información relativa con los usuarios administradores y se podrá invitar a nuevos administradores.
    - **Gestión gestores de casos.** Esta gestión se define por poder gestionar aquellos usuarios que son **gestores de casos** de la aplicación. En ella se podrán ver la información relativa con los usuarios gestores de casos y se podrá invitar a nuevos gestores.
    - **Gestión pacientes.** Esta gestión se define por poder gestionar aquellos usuarios que son **pacientes** de la aplicación. En ella se podrán ver la información relativa con los usuarios que son pacientes.
    - **Gestión cuidadores.** Esta gestión se define por poder gestionar aquellos usuarios que son **gestores de casos** de la aplicación. En ella se podrán ver la información relativa con los usuarios cuidadores.

- **Gestión roles.** Aquí realizamos la gestión de los roles que va a tener la aplicación. Inicialmente nos encontramos con los roles de **administrador**, **gestor de casos**, **paciente** y **cuidador**.
  - **Gestión hospitales.** Aquí realizamos gestiones pudiendo dar de alta, modificar o dar de baja hospitales.
  - **Gestión tratamientos.** Aquí realizamos gestiones pudiendo dar de alta, modificar o dar de baja tratamientos.

- **Gestor casos.** Este rol puede realizar las siguientes operaciones en la aplicación:

- **Gestión citas.** El gestor de casos puede crear, modificar la cita o cancelar una cita para un paciente con ELA. Una vez realizada la operación de crear, cancelar o modificar una cita, llegará una notificación al paciente con la información.
  - **Gestión pacientes.** El gestor de casos puede invitar, ver la información del paciente del cual gestiona el caso o eliminar un paciente fallecido para realizar el derecho al olvido.

- **Paciente**

- **Gestión citas.** En este caso el paciente podrá ver aquellas citas que tenga asociadas y cancelarlas. Una vez cancelada la cita el gestor de casos recibirá una notificación push.
- **Usuarios asociados.** El usuario al registrarse, en la pantalla de usuarios asociados, si no tiene cuidador asignado, podrá enviarle una invitación para que el usuario se registre. También puede ver en esta pantalla la información del gestor de su caso.

#### ■ Cuidador

- **Gestión citas.** En este caso el cuidador podrá ver aquellas citas que tenga asociadas el paciente del que se hace cargo y puede cancelarlas. Una vez cancelada la cita el gestor de casos recibirá una notificación push.
- **Usuarios asociados.** En esta pantalla verá la información del paciente que tenga asignado.

Una vez analizados los requisitos iniciales y ante el tamaño que supone el desarrollo completo de los requisitos, nos ponemos en contacto con la asociación informándoles que para una primera versión nos centraríamos la gestión de los actores que van a interactuar con la aplicación y la gestión de los datos que necesitan los usuarios para realizar la entrega de un proyecto funcional y que pueda ser utilizado por personal sanitario, paciente o cuidador. Bajo nuestro criterio es el más importante crear un producto que se pueda utilizar desde el principio y en futuras y siguientes versiones se iría completando con futuros desarrollos que irán haciendo más robusta y funcional nuestra aplicación.

Algunos ejemplos de futuras versiones son

#### ■ Material ortoprotésico

- **Gestión material.** En este punto se tratará de aquel material que sea de ayuda para facilitar el día a día de los paciente con ELA, como pueden ser gruas o camas.

#### ■ Unidades ELA

- **Gestión unidades ELA.** Aquí el usuario podrá ver aquellos puntos donde se encuentren las unidades de ELA para poder acercarse y consultar o recibir ayuda.

#### ■ Ensayos clínicos

- **Gestión ensayos clínicos.** Se almacenaran y podrán consultar aquellos ensayos clínicos relacionados con la esclerosis.

#### ■ Artículos de Investigación

- **Gestión artículos investigación.** Se almacenaran y podrán consultar aquellos artículos de investigación relacionados con la esclerosis.

Una vez realizada el análisis de los requisitos, nos informamos a cerca de que la **Esclerosis Lateral Amiotrófica (ELA)** [\[17\]](#) es una enfermedad neuromuscular degenerativa que provoca de manera gradual la parálisis muscular y que provoca con el tiempo la muerte del paciente que padece dicha enfermedad.

Tras la investigación y analizando que actores inicialmente van a interactuar con el aplicativo obtenemos tres tipos de usuarios que serían el/la gestor/a de casos que se encargaría de agendar las citas para atender a los pacientes, los cuidadores principales que en un principio serían personas elegidas por los pacientes y finalmente los pacientes que padecen la enfermedad.



---

## 2. Objetivos del proyecto

---

Nuestro proyecto se centra en crear una aplicación móvil para ayudar y dar soporte a personal sanitario, pacientes con ELA o cuidadores principales. A continuación vamos a enumerar los objetivos del proyecto:

- Crear y desarrollar una aplicación multiplataforma amigable.
- Dar soporte a los gestores de casos de pacientes con ELA con la gestión de las citas.
- Dar soporte a personas con ELA con la visualización de las citas.
- Dar soporte a los cuidadores de personas con ELA con la visualización de las citas.
- Facilitar el día a día de los usuarios que usen la aplicación.

### 2.1. Objetivos técnicos

- Desarrollar una aplicación multiplataforma para web y dispositivos Android con API 33 o superior (en futuras versiones implantable en iPhone).
- Adquirir los conocimientos e información necesaria para realizar el proyecto con la arquitectura MVVM (Model-View-ViewModel) [8] para el desarrollo de la aplicación.
- Investigar sobre el framework Flutter [1] desarrollado por Google y crear un proyecto con el resultado de una aplicación multiplataforma con código nativo para distintos dispositivos como en nuestro caso será para Android y web.
- Con los conocimientos obtenidos durante el grado hacer uso de GitHub para el control de versiones y el uso de actions para el despliegue CI/CD en la web y generar la aplicación Android.
- Obtener la habilidad necesaria para hacer uso de la plataforma Firebase de la cual usaremos los servicios disponibles de base de datos, notificaciones, autenticación y hosting para alojar la web.

- Obtener los recursos necesarios para el empleo de LaTeX [10] para realizar la documentación de la memoria y los anexos.
- Obtener los conocimientos en Google Play para tener la posibilidad de distribuir la aplicación móvil.

## **2.2. Objetivos personales**

- Hacer uso de los conocimientos adquiridos durante el grado.
- Investigar sobre el proyecto.
- Emplear frameworks y herramientas novedosas, ágiles y óptimas.
- Ampliar mis conocimientos en el mundo de la programación.

---

## 3. Conceptos teóricos

---

### 3.1. Herramientas proyecto

Para este proyecto no se emplean los conceptos teóricos, ya que la aplicación se desarrolla para facilitar la vida del paciente teniendo a mano las próximas citas médicas que tenga.

A continuación y mediante la tabla se van a mostrar las herramientas que se han empleado para llevar a cabo el proyecto.

Las herramientas usadas para el desarrollo del proyecto son:

Herramientas	App Android	App Web	API REST	BD	Memoria
HTML5		X			
CSS3		X			
BOOTSTRAP		X			
JavaScript		X			
Flutter	X	X	X	X	
Dart	X	X	X	X	
FlutterFlow	X	X	X	X	
Firebase	X	X	X	X	
Github	X	X			
Google Play		X			
T <sub>E</sub> XMaker					X
VersionOne	X	X	X	X	X

Tabla 3.1: Herramientas y tecnologías utilizadas en cada parte del proyecto



---

## 4. Técnicas y herramientas

---

### 4.1. Metodologías

#### Scrum

Scrum [20] es un marco que se emplea para la gestión de proyectos de metodología ágil y que facilita a los equipos la tarea de gestionar y estructurar el trabajo. Para ello, a través de iteraciones (sprints) se realiza una serie de tareas en un periodo de tiempo y se van incorporando al software.

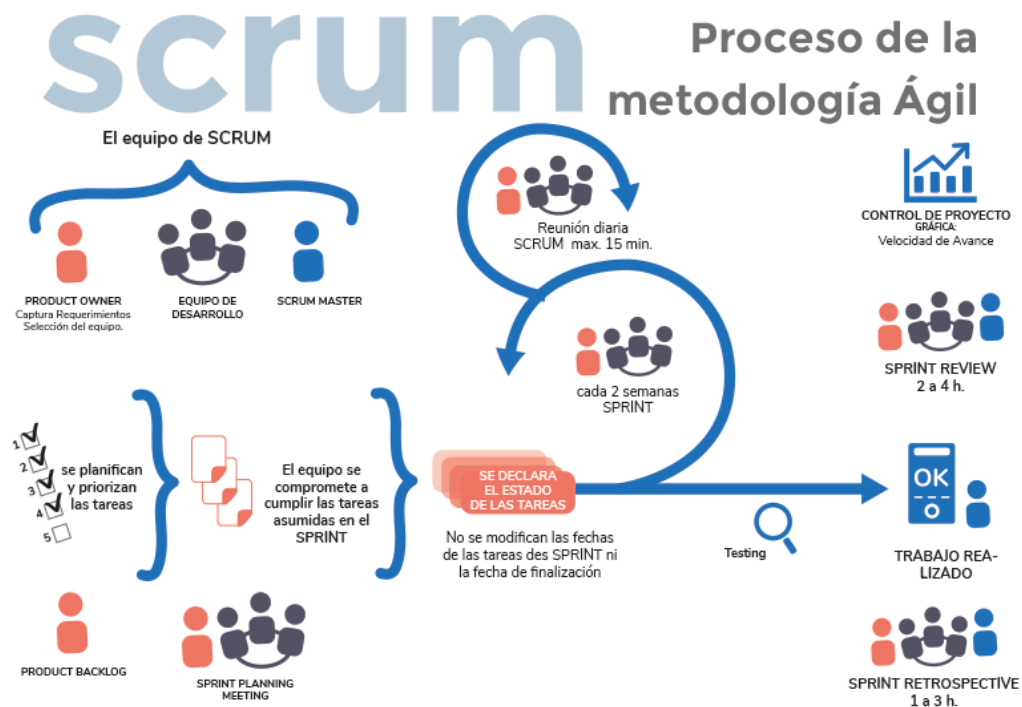


Figura 4.1: Scrum

## Gitflow

Gitflow [5] se define como un flujo de trabajo para la creación de ramas en Git y llevar un control de versiones. En este flujo existen dos ramas principales, master y develop, y otras ramas que pueden ayudarnos en nuestro desarrollo como pueden ser: feature, release o hotfix.



Figura 4.2: Gitflow

## 4.2. Patrón de diseño

### Model-View-ViewModel MVVM

Es un patrón MVVM [8], también conocido como Model View ViewModel que se centra en separar la interfaz del usuario (View) de la parte lógica (Model). La interacción entre la parte lógica y la interfaz del usuario a través del recurso ViewModel.

Algunas de las ventajas al usar este patrón son:

- Fácil desarrollo ya que al poder separar la vista de la lógica varios equipos pueden trabajar simultáneamente en varios componentes.
- Fácil testeo ya que no es necesario utilizar la vista para crear tests para el model o el viewmodel.
- Fácil mantenimiento ya que al realizar la separación de los componentes se crea un código simple y limpio.

Vamos a describir a continuación cada uno de los componentes que forman el MVVM:

- **Model:** es el componente donde se encapsulan los datos de nuestra aplicación. En ella se pueden encontrar validación y lógica de negocio.
- **View:** nos muestra el diseño y la apariencia de nuestra aplicación. En ella se verán los datos pero sin contener nada de la lógica de negocio.
- **ViewModel:** es el componente que enlaza los datos o cambios de estado que puede tener nuestra aplicación.

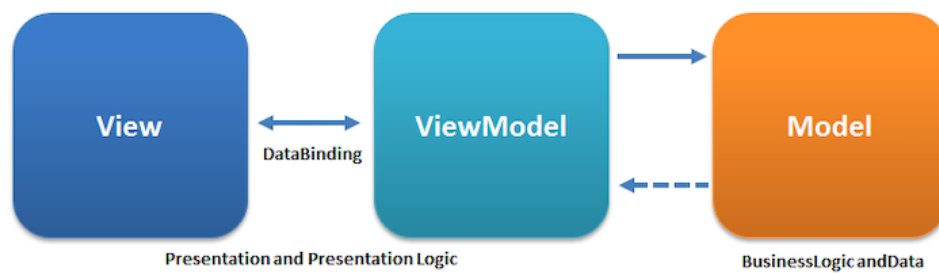


Figura 4.3: Arquitectura limpia

## 4.3. Repositorio

Entre las herramientas consideradas GitHub [2], Bitbucket [3], GitLab [19] se decide utilizar GitHub, porque nos permite alojar proyectos gratuitamente. Además podemos crear documentación a través de wikis, crear tareas, sprints. Al crear el repositorio en GitHub podemos usar el gestor de proyecto que tiene a través de sus paneles y saber el estado en el que está cada issue (tarea). Otra de las ventajas que encontramos con GitHub es que está integrado en múltiples servicios de integración continua.

## 4.4. Control de versiones

Al haber escogido como repositorio GitHub para alojar nuestro proyecto, utilizamos como control de versiones Git [18] que es un sistema distribuido que nos permite realizar nuestro trabajo sin conexión, es ligero y rápido. El crear ramas y hacer los merges es muy rápido y pocas veces se obtienen conflictos. El historial es muy detallado.

## 4.5. Gestión del proyecto

Como gestor de proyectos utilizamos el que viene integrado en GitHub. Con esta herramienta podemos gestionar nuestro proyecto y ver en qué estado está cada tarea, incluso podemos darle prioridad.

## 4.6. Entorno de desarrollo integrado (IDE)

### Flutter



Figura 4.4: Flutter

Flutter [1] es un SDK (Kit de Desarrollo de Software) desarrollado por Google que nos permite crear aplicaciones para varios dispositivos. Hemos elegido esta herramienta porque nos permite crear aplicaciones multiplataforma. Entre alguna de las ventajas destacamos:

- **Compila en nativo** tanto para Android como iOS.
- Mediante el uso de widgets, se pueden **crear interfaces gráficas flexibles**.
- **Rapidez** en el desarrollo.

### Dart

Dart [7] es un **lenguaje de programación moderno**, desarrollado por Google, donde se mezcla la programación orientada a objetos POO con los lenguajes de programación basados en scripts.

### Firebase





Figura 4.5: Firebase

Firebase [6] es una plataforma que nos permite crear aplicaciones web y móviles. Nos permite utilizar un número de herramientas o servicios para facilitarnos a los desarrolladores construir aplicación de forma rápida, segura y escalable. En nuestro proyecto hemos empleado los servicios:

- Autenticación para la gestión de los usuarios.
- Base de datos en la nube con Cloud Firestore que nos facilita la sincronización de los datos almacenados en tiempo real. Hemos decidido emplear Cloud Firestore que es un servicio de base de datos NoSQL porque nos permite una serie de ventajas como son la escalabilidad, eficiencia, flexibilidad, seguridad, entre otras. Otra de las razones de utilizar NoSQL es porque decidimos que cada actor tiene acceso a sus propios datos y aunque estén relacionados entre si, cada uno es dueño de la información que almacena en el aplicativo.
- Notificaciones push con Firebase Cloud Messaging (FCM).
- Hosting para alojar la aplicación web.

## Android Studio

### FlutterFlow

FlutterFlow [4] es una herramienta que nos permite crear aplicaciones insertando los componentes que tiene la herramienta, arrastrando y soltando. Es posible añadir lógica durante la creación de las páginas.

### LaTeX

Entre las distintas herramientas de  $\text{\LaTeX}$  que existen hemos decidido utilizar TeXmaker [21, 10] que es un editor de  $\text{\LaTeX}$  gratuito y multiplataforma. Además, incluye soporte unicode, corrección ortográfica, autocompletado y un visor de PDF integrado.

## Documentación

Para la documentación del trabajo de fin de grado hemos empleado  $\text{\LaTeX}$  [14] que es un sistema que permite componer textos con una alta calidad tipográfica. En el caso de la documentación del repositorio hemos empleado Markdown [16] que es un lenguaje de marcado ligero.

## 4.7. Integración y entregas continuas (CI/CD)

### Github

Github [2] para realizar CI/CD utilizaremos los actions que tiene esta herramienta. A cada push que hagamos contra la rama main esta activará los actions creados y desplegará la aplicación tanto generando la apk de Android como la desplegará en el hosting de Firebase

## 4.8. Herramientas construcción del proyecto

Nuestro proyecto está creado Flutter [1] que nos permite construir aplicaciones nativas multiplataforma a partir de un único código base. Flutter dispone de comandos que permite construir el proyecto y generar distintas aplicaciones como en nuestro caso son la web y Android.

### Librerías

#### Flutter

Flutter dispone de un repositorio oficial para aplicaciones Dart y Flutter <https://pub.dev> que contiene una infinidad de paquetes instalar en nuestro proyecto. Algunos paquetes que cabe destacar son:

- Para securizar los datos de los usuarios que emplean la aplicación **encrypt**.
- Para la gestión de las citas **Syncfusion Flutter Calendar**.
- Para el envío de mails **mailer**.
- Para almacenar nuestras variables de entorno del proyecto **dotenv**.

#### Firebase

Firebase [6] que es una plataforma de Google para desarrollar aplicaciones web y móviles que se encuentra alojada en la nube. Algunas funcionalidades para el funcionamiento de nuestra aplicación que utilizamos de esta plataforma son la autenticación, el hosting para alojar la aplicación, firestore database como base de datos, messaging para las notificaciones push.

Algunos paquetes que necesitamos instalar para poder trabajar con flutter son:

- **Firebase Core**.

- **Firebase Auth.**
- **Firebase Storage.**
- **Cloud Firestore.**
- **Firebase messaging.**



---

## 5. Aspectos relevantes del desarrollo del proyecto

---

Este apartado pretende recoger los aspectos más relevantes del desarrollo del proyecto, pasando por la creación del proyecto, que decisiones se tomaron y la resolución de problemas encontrados y como se solucionaron.

### 5.1. Inicio del proyecto

El proyecto surgió a partir de la necesidad de ayudar a personas diagnosticadas con Esclerosis, tratando inicialmente la EM (Esclerósis Múltiple) y poder hacer su día a día más fácil a través de consejos, atención, ayuda.

Para poder hacer posible esta solución, se crea una aplicación inicialmente para el móvil, donde se incluirán secciones que pueden ayudar a los usuarios que hagan uso de ellas. Tras la asignación del proyecto se inicia un estudio de las posibilidades tecnológicas disponibles para resolver esta necesidad.

Desde el proyecto realizaremos llamadas a una API del cliente que nos proveerá de los datos necesarios para crear una aplicación amigable para los usuarios.

Tras los inconvenientes que surgieron durante el desarrollo de la idea inicial, se decidió enfocar el proyecto a pacientes diagnosticados con ELA (Esclerosis Lateral Amiotrófica), para ello se definió una nueva estructura de datos, empleando documentos que se guardarían en la nube. Esta nueva estructura de datos, se centra en un principio en 3 actores que son los pacientes, los cuidadores principales de los pacientes y gestores/as de casos que gestionarán las citas que necesitan los pacientes para ayudarles a convivir con la enfermedad.

Durante el análisis del proyecto y al tratarse de información sensible de los pacientes, nos proponemos a securizar toda la información de los usuarios que van a utilizar la aplicación por lo que podemos garantizar que la información será cifrada.

### 5.2. Metodologías

Desde el comienzo se realiza una arquitectura limpia para reutilizar los componentes y que sean independientes sin que haya ningún impacto si se realizan

cambios en alguno de ellos. Para la gestión del proyecto, se utiliza el panel que nos proporciona GitHub, usando la metodología ágil Scrum. El equipo al ser un proyecto creado para la realización del trabajo de fin de carrera, está formado por una persona. Por otro lado, el proyecto inicial se alimentaba de una API externa, realizada por un compañero de la universidad, en la que se mantienen reuniones para recabar requisitos técnicos para desarrollar la aplicación. Ante la imposibilidad de continuar con la aplicación, por inconvenientes técnicos, se considera empezar un proyecto nuevo donde seamos nosotros los que controlemos la gestión de los datos de los actores que intervienen en una primera versión del aplicativo.

Para el desarrollo de la metodología ágil se siguieron los siguientes puntos:

- crear iteraciones incrementales (sprints) y revisiones, que esto conlleva a realizar entregas semanales con actualizaciones de la aplicación y revisar aquellos puntos donde hayan surgido problemas.
- una vez terminado el sprint se añade a la rama main que será nuestro producto final. Esto significa que tendremos una versión actualizada de la aplicación añadiendo nuevas funcionalidades.
- los sprints por regla general los hemos hecho de duración semanal.
- para cada sprint se le asocian las tareas más importantes.

### 5.3. Formación

Para abordar la realización y creación del proyecto se necesitaban adquirir una serie de conocimientos que inicialmente eran inexistentes. Entre los conocimientos adquiridos se destacan:

- Formación sobre Flutter [1, 11]. En este punto adquirimos los conocimientos sobre crear elementos en Flutter y poder dar forma a las vistas y funcionalidades que contendrá nuestra aplicación.
- Formación sobre Dart [7]. Nos formamos para obtener los conocimientos necesarios para poder programar y desarrollar la aplicación.
- Formación sobre Firebase [6]. Esta plataforma es la que nos permitira la gestión de los usuarios y almacenar los datos a través de documentos, además, también la gestión de notificaciones push para la creación o cancelación de citas y también para poder alojar la aplicación web.
- Securitizar los datos guardados en la aplicación mediante **AES (Advanced Encryption Standard)** [12] considerado uno de los más seguros y utilizados actualmente que mediante una clave secreta única de cifrado y descifrado.

Además, se profundiza en obtener información a cerca de que es la esclerosis múltiple y que tipos existen. Posteriormente, la aplicación se centra en recabar información de la **esclerosis lateral amiotrófica**.

## 5.4. Desarrollo

Durante el desarrollo de la aplicación nos aparecieron diversos retos de la versión inicial:

- Conexión entre los diferentes componentes.
- Internacionalización de la aplicación.
- Contactar con el cliente ante desconexión de la dirección de la API facilitada inicialmente.
- Contactar con el cliente para resolución de dudas sobre la API.

Durante el desarrollo de la aplicación de la versión final del proyecto nos aparecieron diversos retos:

- Conexión entre los diferentes componentes.
- Contactar con el cliente para resolver dudas para la primera versión.
- Acordar con el cliente que llevaría esta primera versión.
- Trabajar con la plataforma Firebase [6] para la gestión de los datos, notificaciones push.

## 5.5. Publicación

Una vez hubo una versión lista para desplegar, se creó una **cuenta** en **Google Play** y se realizó la subida para una prueba inicial dando posibilidad de descarga por invitación a los testers para poder hacer pruebas y comprobar el funcionamiento de la aplicación. Durante la creación de la **cuenta de desarrollador** de **Google Play**, nos encontramos con el reto de que no podemos verificar nuestra identidad aunque seguimos los pasos que indican durante el proceso. Finalmente y pasando unos días nos verifican la identidad y conseguimos realizar nuestro primer despliegue en **Google Play Console** <https://play.google.com/console/u/0/developers>.

Para próximas versiones, se tratará de crear un despliegue para otros dispositivos, como es iOS.

Cabe recalcar que la aplicación es multiplataforma, por lo que por el momento los usuarios que tengan dispositivos con sistema operativo de iOS podrán acceder a la aplicación a través del navegador web que utilicen.

Este punto lo desarrolleros con más detalle en los anexos, en los que incluiremos que pasos hay que seguir para poder realizar la publicación de la aplicación incluso añadir las claves para poder desplegar una apk en un dispositivo android habilitando el modo **Desarrollador**.





---

## 6. Trabajos relacionados

---

La idea principal realizar una aplicación frontend haciendo uso del backend generado en el TFG **GII\_O\_MA\_22.05 Backend en microservicios para aplicación móvil**. A partir del uso del backend ya implementado se pretende crear una interfaz gráfica multiplataforma para usuarios afectados por la **Esclerosis Múltiple (EM)** y su posible extensión/adecuación para su uso en pacientes con la **Esclerosis Lateral Amiotrófica (ELA)**.

Ante los problemas técnicos observados durante el desarrollo de esta primera fase y bajo el consentimiento de Pedro Renedo Fernández, nos ponemos en contacto con la **Asociación de ELA de Castilla y León** y nos ponemos manos a la obra para crear una aplicación que inicialmente contará con una parte de administrador, que gestionará los usuarios, los hospitales, los tratamientos y roles. Así mismo, también crearemos una aplicación para los usuarios y que tendrá la funcionalidad de la gestión de citas.

Algunas aplicaciones que están relacionadas son:

- **ME – Multiple Esclerosis**. Para realizar un seguimiento y controlar sus síntomas.
- **Cleo**. Centrada en el diálogo directo y con servicio de enfermería.
- **Control EM**. También centrada en el control de la enfermedad.

Podemos decir, que nuestra aplicación será un punto de partida e incluso una aplicación de referencia ya que por las aplicaciones que hemos consultado, en ninguna se tiene en cuenta la gestión de las citas que puede tener el paciente. Otro punto a destacar, que al ser una aplicación multiplataforma, puede abarcar un abanico más amplio a la hora de su uso, es decir, podremos acceder a la aplicación desde cualquier dispositivo o navegador.



---

## 7. Conclusiones y Líneas de trabajo futuras

---

Todo proyecto debe incluir las conclusiones que se derivan de su desarrollo. Éstas pueden ser de diferente índole, dependiendo de la tipología del proyecto, pero normalmente van a estar presentes un conjunto de conclusiones relacionadas con los resultados del proyecto y un conjunto de conclusiones técnicas.

### 7.1. Conclusiones

Las conclusiones que sacamos durante el trayecto del desarrollo de las aplicaciones para el trabajo final de grado, son que una aplicación no siempre que se tiene una idea inicial se consigue llegar a termino y finalizarla exitosamente. Por fortuna, estamos en un mundo en el que tenemos que estar abierto a cambios y a enfrentarnos a retos para poder dar solución a los problemas de los usuarios finales.

A continuación, vamos a enumerar conclusiones obtenidas durante la realización del trabajo:

- Ante los inconvenientes durante el transcurso del proyecto, podemos afirmar que estamos contentos con el resultado obtenido y con el rumbo con el que hemos comenzado. Para empezar esta primera versión se centra en la gestión de las citas que tiene el usuario programadas en el tiempo.
- El haber utilizado el framework Flutter para desarrollar la aplicación nos permite crear una aplicación multiplataforma siendo de código abierto, se integra bien con otras tecnologías, como en nuestro caso es Firebase y una vez adquiriendo la experiencia suficiente facilita el desarrollo.
- Se emplean los conocimientos utilizados durante el grado para el desarrollo de la aplicación.

### 7.2. Líneas de trabajo futuras

Como en todas las aplicaciones se parte de una versión inicial y esta aplicación puede seguir creciendo para ir incluyendo nuevas funcionalidades y desarrollos que facilitarán el día a día de los usuarios.

- Crear opción para ver el material ortoprotésico
- Crear opción para ver el listado de las unidades de ELA disponibles
- Crear opción para ver el listado de las los ensayos clínicos
- Crear opción para ver el listado de las artículos de investigación
- Crear opción para mantener una conversación con la gestora de casos para agendar una cita.

---

## Bibliografía

---

- [1] aurestic. ¿qué es flutter? <https://aurestic.es/que-es-flutter/>.
- [2] Gustavo B. Qué es github y cómo empezar a usarlo. <https://www.hostinger.es/tutoriales/que-es-github>, 2023.
- [3] Atlassian Bitbucket. Breve presentación de bitbucket. <https://bitbucket.org/product/es/guides/getting-started/overview#a-brief-overview-of-bitbucket>, 2024.
- [4] Mateus Castelo. Flutterflow | la revolución en la creación de apps nocode. <https://nocodestartup.io/es/flutterflow/#:~:text=Conclusi%C3%B3n%20sobre%20FlutterFlow-,%C2%BFQu%C3%A9%20es%20FlutterFlow%3F,e%20inserta%201%C3%B3gica%20en%20ella>.
- [5] Aula de Software Libre. Flujo de trabajo con git (git flow). <https://aulasoftwarelibre.github.io/taller-de-git/gitflow/>.
- [6] Firebase. <https://firebase.google.com/?hl=es-419>.
- [7] Juan Gabriel Gomila. ¿qué es dart? historia, características y ventajas de aprenderlo. <https://cursos.frogamesformacion.com/pages/blog/que-es-dart>, 2023.
- [8] Flutter Guru. Flutter mvvm architecture. <https://blog.devgenius.io/flutter-mvvm-architecture-with-provider-a81164ef6da6>, 2022.
- [9] MedlinePlus. <https://medlineplus.gov/spanish/multiplesclerosis.html>.
- [10] Texmaker. <https://www.xmlmath.net/texmaker/>.
- [11] Udemy. Flutter desde cero - guía completa con arquitectura limpia. <https://www.udemy.com/course/flutter-desde-cero-darwin-morocho/?couponCode=2021PM20>.
- [12] Wikipedia. Advanced encryption standard. [https://es.wikipedia.org/wiki/Advanced\\_Encryption\\_Standard](https://es.wikipedia.org/wiki/Advanced_Encryption_Standard).
- [13] Wikipedia. Esclerorisis. <https://es.wikipedia.org/wiki/Esclerosis>.

- [14] Wikipedia. Latex — wikipedia, la enciclopedia libre. <https://es.wikipedia.org/w/index.php?title=LaTeX&oldid=84209252>, 2015. [Internet; descargado 30-septiembre-2015].
- [15] Wikipedia. [https://es.wikipedia.org/wiki/Esclerosis\\_m%C3%BAltiple](https://es.wikipedia.org/wiki/Esclerosis_m%C3%BAltiple), 2024.
- [16] Wikipedia. <https://es.wikipedia.org/wiki/Markdown>, 2024.
- [17] Wikipedia. Esclerosis lateral amiotrófica. [https://es.wikipedia.org/wiki/Esclerosis\\_lateral\\_amiotr%C3%B3fica](https://es.wikipedia.org/wiki/Esclerosis_lateral_amiotr%C3%B3fica), 2024.
- [18] Wikipedia. Git. <https://es.wikipedia.org/wiki/Git>, 2024.
- [19] Wikipedia. Gitlab. <https://es.wikipedia.org/wiki/GitLab>, 2024.
- [20] Wikipedia. Scrum (desarrollo de software). [https://es.wikipedia.org/wiki/Scrum\\_\(desarrollo\\_de\\_software\)](https://es.wikipedia.org/wiki/Scrum_(desarrollo_de_software)), 2024.
- [21] Wikipedia. Texmaker. <https://es.wikipedia.org/wiki/Texmaker>, 2024.