

**Trabajo De Fin De Grado**

David Corredor Miguel, Antonio Castillo y Francisco León

Tutor: Aníbal Martin

**Curso 2023/24**

Volando Voy



**Grado Superior**

**DAM**

**Contenido**

[**1** **Agradecimientos/Dedicatoria** 3](#_Toc168592678)

[**2** **Resumen** 3](#_Toc168592679)

[**3** **Abstract (Resumen En inglés)** 4](#_Toc168592680)

[**4** **Introducción** 5](#_Toc168592681)

[**5** **Alcance funcional del sistema** 6](#_Toc168592682)

[**6** **Diseño Técnico** 6](#_Toc168592683)

[6.1 Diagrama De Clases 6](#_Toc168592684)

[6.2 Esquema De Base De Datos 7](#_Toc168592685)

[6.3 Diseño De La Interfaz De Usuario 7](#_Toc168592686)

[6.3.1 Login 8](#_Toc168592687)

[6.3.2 Inicio 8](#_Toc168592688)

[6.3.3 Localidades 8](#_Toc168592689)

[6.3.4 Detalles Localidad 8](#_Toc168592690)

[6.3.5 Retos 8](#_Toc168592691)

[6.3.6 Crear Localidad 8](#_Toc168592692)

[6.3.7 Crear Retos 8](#_Toc168592693)

[**7** **Tecnología Utilizada** 8](#_Toc168592694)

[7.1 MAUI 8](#_Toc168592695)

[7.2 .NET 9](#_Toc168592696)

[7.3 Supabase 10](#_Toc168592697)

[7.4 GitHub 11](#_Toc168592698)

[7.5 Visual Studio 11](#_Toc168592699)

[7.6](#_Toc168592700) [API de Google](#_Toc168592000)

[**8** **Presupuesto** 11](#_Toc168592701)

[**9** **Planificación** 12](#_Toc168592702)

[9.1 Investigación y Análisis 12](#_Toc168592703)

[9.2 Preparación Del Entorno De Trabajo 13](#_Toc168592704)

[9.3 Definir Requisitos y Etiquetas Del Formulario Aplicación 13](#_Toc168592705)

[9.4 Creación De La Aplicación 13](#_Toc168592706)

[9.5 Instalación y Configuración 13](#_Toc168592707)

[9.6 Generar Documentación 13](#_Toc168592708)

[9.7 Diagrama 13](#_Toc168592709)

[**10** **Manual De Instalación** 13](#_Toc168592710)

[**11** **Manual De Usuario** 13](#_Toc168592711)

[11.1 Si No Has Iniciado Sesión 13](#_Toc168592712)

[11.2 Si Has Iniciado Sesión 14](#_Toc168592713)

[**12** **Conclusiones** 14](#_Toc168592714)

[**13** **Bibliografía** 14](#_Toc168592715)

# **Agradecimientos/Dedicatoria**

Queremos agradecer al Centro De FP Juan XXIII de Alcorcón y a sus profesores que nos han impartido clases y enseñando su conocimiento sobre el mundo del desarrollo, la programación multiplataforma, el diseño de aplicaciones.

Gracias a ellos, y a sus diferentes clases y asignaturas que hemos cursado a lo largo de los dos años de este el ciclo, hemos podido descubrir un oficio en el que tenemos pasión por trabajar y seguir aprendiendo y formándonos en los diferentes campos del gran mundo de la informática y el desarrollo multiplataforma.

Además de dar las gracias a los profesores del centro, también tenemos que dárselas a todos los compañeros que hemos compartido este camino y nos han permitido comprobar cómo es el trabajo en equipo dentro de este sector y lo importante que es unir fuerzas junto a compañeros de trabajo o de proyecto para conseguir un desarrollo más completo y menos problemático en el camino.

También nos gustaría agradecer a nuestra familia y amigos que han participado en el desarrollo de esta aplicación y de este trabajo final de grado, desde las personas que nos han dado su opinión acerca de diferentes aspectos de la aplicación, así como

las que me han ayudado a probar y confirmar que nuestra aplicación es funcional y totalmente válida para ser catalogado como una aplicación oficial de Volando Voy.

Por último, nos gustaría agradecer a las personas que valorarán este proyecto, el dedicar parte de su tiempo a conocer un trabajo que ha sido nuestro principal proyecto durante los meses anteriores a su realización y del que personalmente sentimos orgullo y satisfacción de mostrarlo al público tras finalizarlo.

# **Resumen**

Este Proyecto Fin De Grado sirve para mostrar lo que hemos aprendido en Desarrollo de Aplicaciones Multiplataforma. Es una aplicación que muestra al usuario diferentes sitios turísticos (pueblos, ciudades) usando una base de datos online (Supabase) para que se pueda utilizar en varios dispositivos, tanto Windows como Android, y presentando estos en una aplicación móvil para que los usuarios puedan interactuar con ella y con los contenidos en cualquier lugar.

Incluye un ranking de los comentarios de los sitios mejor valorados. En ella el usuario puede crearse una cuenta (nombre de usuario, imagen de usuario, correo y contraseña) para interactuar con ella, comentar sobre los sitios y poner valoraciones.

Si no hay ningun usuario en la base de datos online el usuario que se cree será admin. La diferencia es que los usuarios que sean admin pueden acceder a las ventanas de Crear Localidad y Crear Retos.

Tiene un menú lateral con las ventanas, el nombre de usuario y su imagen. Si pulsa en Cerrar Sesión vuelve al Login.

En la ventana de Inicio aparecen el logo, una descripción del programa de televisión de Jesús Calleja “Volando Voy” y un mapa que muestra el estudio de Mediaset España.

En la ventana de Localidad aparecen el logo, una lista con la imagen de la localidad y su nombre. Si pulsa en el nombre de la Localidad le lleva a la ventana de Detalles Localidad.

En la ventana de Detalles Localidad aparecen el logo, dos columnas, una columna con la imagen de la localidad y su provincia y la otra columna con la imagen de la comunidad autónoma y su comunidad autonomía y los 3 mejores comentarios (mejor valoración). Si pulsa en Mostrar Más Comentarios aparecerán todos los comentarios de la Localidad.

En la ventana de Retos aparecen un desplegable con el nombre de la Localidad. Si pulsa en el nombre de la Localidad le mostrará sus Retos.

# **Abstract (Resumen En inglés)**

This Final Degree Project serves to show what we have learned in Multiplatform Application Development. It is an application that shows the user different tourist sites (towns, cities) using an online database (Supabase) so that it can be used on various devices, both Windows and Android, and presenting these in a mobile application so that users can interact with it and with the content anywhere.

It includes a ranking of the comments of the best rated sites. In it the user can create an account (username, user image, email and password) to interact with it, comment on the sites and give ratings.

If there is no user in the online database, the user that is created will be admin. The difference is that users who are admin can access the Create Locality and Create Challenges windows.

It has a side menu with the windows, the username and their image. If you click on Close Session you return to Login.

In the Home window, the logo appears, a description of Jesús Calleja's television program “Volando Voy” and a map showing the Mediaset España studio.

In the Locality window, the logo appears, a list with the image of the locality and its name. If you click on the name of the Locality, it takes you to the Locality Details window.

In the Locality Details window, the logo appears, two columns, one column with the image of the locality and its province and the other column with the image of the autonomous community and its autonomous community and the 3 best comments (best rating). If you click on Show More Comments, all the comments for the Locality will appear.

In the Challenges window, a drop-down menu appears with the name of the Locality. If you click on the name of the Locality, it will show you, its Challenges.

# **Introducción**

Este trabajo final de grado ha tenido el objetivo de mostrar todos los conocimientos

que han sido adquiridos durante la realización del ciclo, de forma que con ellos y a través

de nuevos conocimientos aprendidos de forma autodidacta con investigaciones en

diferentes aplicaciones MAUI encontradas en Internet se haya podido crear una aplicación totalmente funcional y que, tras subirlo a un hosting, se pueda utilizar de forma funcional por cualquier persona ajena a los creadores de esta.

Esta aplicación muestra al usuario diferentes sitios turísticos (pueblos, ciudades) y un ranking de los sitios mejor valorados. Además, permite comentar sobre los sitios y poner valoraciones.

Volando Voy es un programa de aventuras de Jesús Calleja (emitido desde el 19 de julio de 2015 hasta el 16 de octubre de 2023 en los canales de Mediaset España, en Cuatro y reposiciones en BeMad) que junto a ciudadanos españoles recorrerá toda la geografía española desde un helicóptero (Robinson R44). Los habitantes de diferentes regiones españolas serán testigos en primera persona de las vistas más espectaculares de sus localidades que siempre han tenido a su alcance y nunca han podido disfrutar. Investigaciones y estudios cartográficos en 3D o desarrollo de nuevas técnicas de salvamento marítimas serán dos de las muchas cosas que Jesús Calleja y su equipo completarán con éxito suponiendo un gran avance para los locales de cada pueblo. Al final de cada entrega, el equipo del programa y todos los vecinos de los pueblos organizarán una fiesta en torno a la vista del documental previamente grabado teniendo así el privilegio de poder disfrutar del programa y compartir las emociones y anécdotas en torno a los vecinos de la localidad. En algunos programas, famosos, ayudarán a Jesús Calleja a conseguir los retos propuestos por el programa.

# **Alcance funcional del sistema**

Se va a analizar el alcance funcional del proyecto y con ello la funcionalidad que este

tiene y que podría aportar a los clientes que visiten la aplicación. La aplicación da soporte las funcionalidades:

* Log On (Iniciar Sesión y Crear Cuenta).

# **Diseño Técnico**

En este punto se detallará elementos del diseño técnico de la aplicación. Se expondrán elementos como el diagrama de clases utilizado para desarrollar el modelo de clases que organiza el desarrollo de la programación, así como el esquema que sigue la base de datos y un resumen del diseño de la interfaz de usuario.

Este proyecto tiene un diseño técnico y una arquitectura enfocada en el patrón de arquitectura software Modelo-Vista-Controlador (MVC). En el que se ha separado la representación de la información y la interacción del usuario, en el modelo y vista, respectivamente, y los controladores que son los que unen los modelos con la vista.

En este desarrollo se entiende como vista cada una de las páginas que van a ser mostradas al usuario en la interfaz web, los modelos son la unión del desarrollo con la base de datos, en estos declaramos clases con los métodos que obtendrán la información de la base de datos y que devolverán la información obtenida al controlador, que será el encargado de administrarla y enviarla a la vista para ser mostrada al usuario final. Por lo que cada controlador está enfocado a una página diferente, y los modelos se dividen en tantos como tablas existan en la base de datos.

Tenemos otro componente implementado en el desarrollo, los traits, que son un mecanismo usado para la reutilización de código, en estos declaramos y desarrollamos funciones que, al igual que los modelos, también hacen consultas para obtener información de las bases de datos.

Pero mientras que los modelos solo pueden ser vistos por sus controladores, los traits pueden ser vistos y por lo tanto sus métodos puedes ser utilizados por todos los controladores. De esta forma podemos reutilizar métodos y consultas a la base de datos que van a ser necesitados desde diferentes controladores.

## Diagrama De Clases

Se va a mostrar el diagrama de clases utilizado para el diseño técnico del desarrollo

funcional de la aplicación. En este diagrama podemos observar las clases referentes a los modelos y traits existentes en nuestro proyecto y que nos permiten recoger la información de la base de datos, guardarla en variables, en el caso de este proyecto para que administre la información recibida por los modelos y mostrarlas sus vistas.

Este desarrollo ha utilizado MAUI como lenguaje para trabajar en el lado del servidor. Este lenguaje está caracterizado por ser un lenguaje no tipado o débilmente tipado, lo que este TFG quiere decir es que, no hace falta declarar que tipo tendrá la variable que va a ser declarada, puesto que en cualquier momento del desarrollo puede cambiar de tipo sin problema.

Por lo que las propiedades, métodos y parámetros, que veremos en este diagrama de clases que hemos usado como ejemplo y demostración de nuestro desarrollo, carecerán de un tipado concreto.

La siguiente imagen es el diagrama de clases del modelo, las clases están representadas por un cuadrado, en el que se puede ver el nombre de la clase, los atributos, y los métodos con sus parámetros que han sido declarados en esta.

## Esquema De Base De Datos

Ahora se va a mostrar un esquema de la base de datos que se ha diseñado

por el desarrollo de esta aplicación. Se ha buscado una normalización de las tablas para minimizar la redundancia de los datos y así poder facilitar su gestión.

La creación de las tablas se ha basado principalmente en los datos que eran importantes tener guardados para posteriormente poder ser administrados. Lo más importante era poder administrar los datos del panel de administración que era la información que había que administrar para que el sitio web funcionara correctamente.

En estas tablas queda recogida la información que es totalmente necesaria guardar ya sea por mantener su privacidad o porque sea necesaria para la administración del sitio web de manera segura y correcta.

La siguiente figura es el esquema de la base de datos que se ha usado y las relaciones que las tablas mantienen entre sí.

## Diseño De La Interfaz De Usuario

Para el diseño de la interfaz de usuario de la aplicación se ha tenido en cuenta principalmente un diseño sencillo y moderno, en el que no haya un exceso de detalles que causen una sensación de desorden visualmente a los usuarios que visiten la aplicación.

Se ha querido buscar el minimalismo y la simpleza por delante de diseños con un alto nivel de detalles, dónde lo que resalte principalmente sea la información que se quiere Volando Voy enseñar al usuario para que este entienda el alcance y el significado de la aplicación y se quede con lo que la aplicación ofrece y no con los detalles visuales que esta puede tener.

En cuanto a la paleta de colores que se ha utilizado, principalmente se ha usado una gama de colores rosados que encajan con los usados en el logo de Volando Voy y negros sobre blancos para buscar la sencillez mencionada anteriormente.

Hay que dividir el diseño en dos partes, la aplicación principal para todos los usuario, y, por otra parte, la aplicación principal para todos los admin.

## Login

## Inicio

## Localidades

## Detalles Localidad

## Retos

## Crear Localidad

## Crear Retos

# **Tecnología Utilizada**

## MAUI

Hemos decidido usar MAUI para tener un código multiplataforma, para una mayor compatibilidad al ejecutar la aplicación en diferentes entornos. Sirve para crear aplicaciones móviles y de escritorio nativas con C# y XAML que se pueden ejecutar en Android, iOS, macOS y Windows desde una sola base de código compartida. Tiene controles de interfaz de usuario recopilados para mejorar el rendimiento y la extensibilidad.

Uno de sus objetivos clave es implementar la mayor parte de la lógica de la aplicación y el diseño de la interfaz de usuario en una única base de código. Nos ha resultado útil por usar una base de código compartida única, permite compartir el diseño de la interfaz de usuario entre plataformas y también compartir código, pruebas y lógica de negocios entre plataformas. Sus **Características** son:

* Crear aplicaciones multiplataforma con un solo proyecto, pero puede agregar recursos y código fuente específicos de la plataforma si es necesario.
* Un motor de diseño elaborado para diseñar las páginas.
* Varios tipos de páginas para crear tipos de navegación enriquecidos.
* Compatibilidad con XAML y el enlace de datos, para obtener patrones de desarrollo más elegantes y fáciles de mantener.
* Crear controladores personalizados para mejorar la forma en que se presentan los elementos de la interfaz de usuario.
* API multiplataforma para acceder a las características de dispositivos nativos: que permiten a las aplicaciones acceder a características del dispositivo, como el GPS que hemos usado para el mapa.
* Funcionalidad de gráficos multiplataforma, que proporciona un lienzo de dibujo que admite formas e imágenes de dibujo y pintura, operaciones de redacción y transformaciones de objetos gráficos.
* Un único sistema de proyecto que usa varios destinos para tener como destino Android, iOS, macOS y Windows.
* Recarga activa de .NET, para modificar tanto el XAML como el código fuente administrado mientras se ejecuta la aplicación y luego ver las modificaciones sin volver a generar la aplicación.

## .NET

Hemos decidido usar el lenguaje de programación aplicación .NET porque permite interactuar con los usuarios y utilizar sistemas de Bases de Datos. Además de ser una plataforma de aplicaciones segura, confiable y de alto rendimiento.

C# es el lenguaje de programación que usa, es fuertemente tipado y tiene seguridad de tipos, presenta simultaneidad integrada y administración automática de memoria. Sus componentes principales son:

* **Entorno de ejecución:** que ejecuta código de aplicación.
* **Bibliotecas:** que proporciona funcionalidad de la utilidad como análisis de JSON.
* **Compilador:** que compila código fuente de C# (y otros lenguajes) en código ejecutable (entorno de ejecución).
* **SDK y otras herramientas:** que permiten la creación y supervisión de aplicaciones con flujos de trabajo modernos.
* **Pilas de aplicaciones:** como ASP.NET Core y Windows Forms, que permiten escribir aplicaciones.

## Supabase

Hemos decidido usar Supabase como gestor de Base de Datos online. Es una plataforma BaaS (Backend as a Service) alojada en la nube que da a desarrolladores una amplia gama de herramientas para crear y gestionar servicios backend. Esto permite subcontratar funciones y desarrollar las aplicaciones de manera ágil, sin tener que preocuparse por las tareas relacionadas con el lado del servidor.

Es una alternativa de código abierto a Firebase de Google, pero con una interfaz intuitiva. No necesita instalaciones para poder usarla, solo crear una cuenta para comenzar a desarrollar un proyecto. Ofrece todos los servicios y herramientas de backend necesarias para crear una aplicación escalable y segura: gestión de base de datos, autenticación, almacenamiento de archivos, generación automática de APIs y actualizaciones en tiempo real, entre otros. Sus **Ventajas** son:

* Su interfaz intuitiva puede significar un ahorro considerable de tiempo e inversión a para desarrollar una aplicación.
* **Código abierto:** se tiene acceso completo al código fuente, por lo tanto, este puede ser personalizado según las necesidades específicas.
* **Comunidad en crecimiento:** cuenta con una comunidad de colaboradores que crece diariamente, donde hay muchos usuarios dispuestos a brindar soporte para encontrar la solución a cualquier problema.
* **Base de datos relacional Postgres:** utiliza (PostgreSQL), la cual es muy flexible y permite crear aplicaciones en tiempo real ofreciendo, al mismo tiempo, mayor soporte para consultas complejas y de integración de datos.
* **Múltiples opciones de implementación:** es posible implementar la aplicación o servicio en la nube de manera muy fácil y rápida a través de su interfaz de línea de comandos o utilizando su panel de control.
* **No tiene dependencia del proveedor:** a diferencia de otros BaaS, que pueden sufrir bloqueos por parte de su proveedor, con Supabase este problema no existe, ya que debido a que es de código abierto, esta no depende de limitaciones de terceros.

## GitHub

Hemos decidido usar GitHub como repositorio de archivos. Es un servicio basado en la nube que aloja un sistema de control de versiones (VCS) llamado Git que permite a los programadores colaborar y realizar cambios en proyectos compartidos, a la vez que mantienen un seguimiento detallado de su progreso. Los repositorios los hemos usado para guardar archivos especialmente los lenguajes de programación de MAUI, editarlos y ponerlos en común.

- ***Github Pages*** = Además GitHub tiene este apartado que hemos utilizado para guardar nuestra página web que tenemos para poder instalar nuestra aplicación.

Hemos elegido GitHub Pages debido a que es gratis y las facilidades que no da de poder tener la web en un servidor sin tener que pagar ni tener que crear nosotros el servidor.

## Visual Studio

Hemos decidido usar Visual Studio como entorno de desarrollo. Es una plataforma de lanzamiento creativa que puede utilizar para editar, depurar y compilar código de una aplicación. Además del editor y depurador estándar que ofrecen la mayoría de IDE, incluye compiladores, herramientas de completado de código, diseñadores gráficos y muchas más funciones para mejorar el proceso de desarrollo de software.

## API de Google

Para nuestro proyecto hemos utilizado las siguientes API de Google accediendo a ellas mediante la consola para desarrolladores de Google :

7.6.1 Maps SDK for Android

Hemos utilizado esta API para mostrar los mapas que se ven en la aplicación para mostrar localizaciones cuando se esta ejecutando la aplicación en un dispositivo Android.

7.6.2 Bing Maps

Esta API no es de Google pero la mencionamos aquí debido a que para poder mostrar un Mapa cuando la aplicación se ejecuta en Windows la API anterior no nos sirve y la que encontramos que mejor funcionaba era esta.

7.6.3 Places API (new)

Esta API nos ha servido para varias cosas en nuestra aplicación ya que puede ofrecer muchas funciones como conseguir coordenadas de un sitio, conseguir lugares de interés , conseguir fotos de cada lugar etc.… Se ha usado su versión **new** en vez de la normal ya que ofrece el doble de lugares (100M a 200M) que la versión normal.

Las funciones que hemos utilizado nosotros han sido las siguientes :

- ***Autocompletar*** : Esta función nos ayuda en la cuando insertamos las localidades, hace peticiones a la API de places para encontrar localidades que su nombre se parezca a lo que escribimos. Devuelve una lista de localidades que son las que se ven en la aplicación mientras se va escribiendo.

- ***Place Details*** : Esta función de la API se encarga de darnos datos de la localidad elegida como la Provincia a la que pertenece , la Comunidad Autónoma y las coordenadas de Longitud y Latitud que las necesitamos para mostrar en el mapa donde esta la localidad y para los lugares de interés.

- ***Nearby Search*** : Esta función de la API se encarga de buscar lugares de interés de la localidad. Para ello tenemos que pasarle a la consulta de la API las coordenadas que conseguimos con Place Details y con esas coordenadas se encargar de buscar lugares de interés en un radio que le puedes pasar ademas de poder filtrar tipos de lugares descartando algunos que no quieras que salgan.

## SMTP Gmail

Hemos decidido usar el SMTP (Simple Mail Transfer Protocol) de Gmail para enviar correos de forma gratuita mediante el puerto 587 con TLS. Esto se puede conseguir en tu cuenta de Google creando una ***contraseña de aplicación*** que se usa para poder autenticarte en el servidor SMTP de Gmail. En nuestra aplicación la utilizamos para poder enviar correos para la confirmación de registro y para la recuperación de contraseña donde enviamos un código generado en nuestra aplicación.

# **Presupuesto**

Vamos va a elaborar un presupuesto indicando cuánto costaría el desarrollo de esta aplicación a un cliente que lo solicitara.

Este presupuesto se elabora basándose en los costes que pueden salir al tener que comprar las licencias de determinados lenguajes o plantillas y otros aspectos para tener en cuenta como el salario que hay que pagar a desarrollador por las horas trabajadas, el alquiler de equipos o de un espacio de trabajo y el coste que puede suponer la contratación de un hosting y de un dominio en el caso de querer subir el proyecto a internet.

En cuanto a las tecnologías usadas, todas ellas son de código abierto, por lo que son tecnologías gratuitas que los desarrolladores han puesto libres para el uso de la comunidad. Las únicas que no cumplen con eso son las APIs de Google, que en nuestro

En cuanto al salario del trabajador por hora trabajada, se le pagaría 7,50 la hora en un contrato de 40 horas semanales por lo que tendría un salario base de 1200 € brutos al mes.

El alquiler de equipo constaría de un portátil con un coste de 29 € al día, contratado a una compañía externa de alquileres de equipos informáticos, lo que supondría un coste de 580 € mensuales.

Por último, el alquiler de un espacio de trabajo que será usado como oficina por el trabajador o trabajadores que participen en el proyecto tendría un coste de 100 € al mes, un espacio de trabajo compartido en la zona de Alcorcón que permitiría a los trabajadores tener su espacio de trabajo personal.

Incluimos una tabla con los cálculos de presupuesto mensuales teniendo en cuenta los gastos principales mencionados anteriormente, para la creación del Volando Voy (TFG) presupuesto principal que poder dar a un cliente que contrate los servicios para la creación de este proyecto de aplicación multiplataforma.

|  |  |
| --- | --- |
| **PRESUPUESTO INICIAL DE LA APLICACIÓN** | |
| **LICENCIAS DE DESARROLLO** | 10 € |
| **SALARIO DE LOS TRABAJADORES (3 TRABAJADORES)** | 3600 € |
| **ALQUILER DE EQUIPO DE TRABAJO (3 TRABAJADORES)** | 1740 € |
| **ALQUILER DE ESPACIO DE TRABAJO** | 100 € |
| **TOTAL** | 5450 € |

# **Planificación**

El proyecto se desarrolló al final del tercer semestre del curso 2023-2024, y se realizaron aproximadamente 100 horas. Se creó un plan de trabajo y se dividió en 7 fases:

## Investigación y Análisis

Es la búsqueda de información y análisis de los posibles lenguajes de programación y frameworks que se podrán usar en el proyecto. Se analizará ventajas e inconvenientes de cada uno y con ejemplos básicos para ver su funcionamiento. Una vez analizada la información buscada, se realizará una inmersión, para elegir la que mejor se adapte a nuestras necesidades.

## Preparación Del Entorno De Trabajo

Es instalar y probar el entorno de desarrollo, para luego realizar pruebas básicas. Útil para un acercamiento inicial con respecto al desarrollo del proyecto.

## Definir Requisitos y Etiquetas Del Formulario Aplicación

Se definen los requisitos iniciales de la aplicación. También se definen las etiquetas que se van a usar dentro de la plantilla del formulario.

## Creación De La Aplicación

* Diseño MVC: Una de las fases más importantes del desarrollo es crear la arquitectura del programa y se hará una estructura básica de nuestra aplicación.
* Implementación: Después de saber requisitos y diseño de la aplicación, implementar el sistema cumpliendo con las especificaciones definidas (esta tarea la más larga).
* Pruebas: Una vez terminado el proyecto se realizarán pruebas con usuarios reales, pudiendo detectar los posibles errores y corregirlos.

## Instalación y Configuración

Se instalará y configurará la aplicación en los sistemas de la empresa, estos son: sistema de preproducción y de producción.

## Generar Documentación

Consiste en redactar varios documentos, tales como: los informes de reunión, el plan de trabajo, el estado del arte, la memoria de seguimiento, la memoria final, la presentación y el manual de usuario.

## Diagrama

Consiste en redactar varios documentos, tales como: los informes de reunión, el plan de trabajo, el estado del arte, la memoria de seguimiento, la memoria final, la presentación y el manual de usuario.

# **Manual De Instalación**

1. Para ver este proyecto de tienda aplicación es instalar Visual Studio con Desarrollo de Interfaz de Usuario de Aplicaciones Multiplataforma para incluir MAUI.
2. Instalar el emulador en Visual Studio.
3. Después abrir la solución del proyecto, iniciar el emulador pulsando la flecha verde.
4. Por último, iniciar la aplicación pulsando la flecha verde.

# **Manual De Usuario**

## Si No Has Iniciado Sesión

1. Al hacer click en Iniciar Sesión aparecerá para iniciar sesión.
2. Al hacer click en Volver A Inicio volverá a la página de Inicio (Sin Sesión).
3. Al hacer click en Crear Usuario aparecerá para crear un nuevo usuario.
4. Después de escribir el nombre de usuario, contraseña. Haz click en Registrarse para crear el usuario.
5. Si se ha creado el usuario correctamente haz click en Iniciar Sesión para volver al formulario para iniciar sesión.
6. Por último, introduce tu nombre de usuario y contraseña para iniciar sesión.

## Si Has Iniciado Sesión

1. Al hacer click en Cerrar Sesión volverá a la página de Inicio (Sin Sesión).

# **Conclusiones**

Para concluir, este proyecto, como estudiantes, nos ha dado la oportunidad de poder demostrar nuestros conocimientos aprendidos durante el ciclo de Desarrollo de Aplicaciones Multiplataforma realizado.

Tras dos años estudiando el sector, las tecnologías usadas y la forma de unirlas para hacer un desarrollo válido y profesional, me ha permitido poder aprovechar todos estos conocimientos aprendidos para la realización de este desarrollo y sobre todo para poder participar en el proyecto Volando Voy aportando nuestra capacidad de poder crear una aplicación que permita a los fans del programa poder tener en una aplicación totalmente funcional un sitio donde encontrar los sitos del programa, sus retos y además poder crear comentarios y valoraciones de los sitios.

La realización de un proyecto final de grado al término de los estudios te permite comprobar que has asentado los conocimientos aprendidos y que además puedes ser totalmente capaz de utilizarlos para participar en un proyecto y comenzar un desarrollo que te permita experimentar una satisfacción laboral y profesional al poder demostrar que tienes la capacidad de trabajar como programador, diseñador o desarrollador multiplataforma entre otras laborales cuyos conocimientos han sido enseñados en la formación profesional.

Para finalizar este trabajo de fin de grado, quiero añadir que he podido experimentar una satisfacción profesional al mostrar los conocimientos aprendidos en este desarrollo multiplataforma y poder formar parte de Volando Voy como proyecto, sintiendo una buena culminación profesional y estudiantil al finalizar nuestro proceso de formación profesional como estudiante de desarrollo multiplataforma.

# **Bibliografía**

En este punto para finalizar el trabajo, se añaden algunas de las páginas en las que hemos podido recabar información, documentación y conocimientos que me nos han sido prácticos para la realización del desarrollo y del trabajo en alguno de sus ámbitos.

**Webs de información:**

A continuación enlaces de webs donde hemos recabado infomación que nos ha ayudado en la aplicación :

1. https://stackoverflow.com/

2.[Documentación Places API](https://developers.google.com/maps/documentation/places/web-service/op-overview?hl=es-419)

**Webs para la Aplicación:**

1. [https://www.w3schools.com](https://www.w3schools.com/)
2. [https://www.github.com](https://www.github.com/)

**Imágenes Web de Descarga:**

1. <https://www.iconsdb.com/lime-icons/android-6-icon.html>
2. <https://www.iconsdb.com/white-icons/os-windows8-icon.html>

**Imágenes Aplicación:**

1. <https://www.iconsdb.com/black-icons/home-7-icon.html>
2. <https://www.iconsdb.com/white-icons/home-7-icon.html>
3. <https://www.iconsdb.com/black-icons/visible-icon.html>
4. <https://www.iconsdb.com/white-icons/visible-icon.html>
5. <https://www.iconsdb.com/black-icons/invisible-icon.html>
6. <https://www.iconsdb.com/white-icons/invisible-icon.html>
7. <https://www.iconsdb.com/black-icons/logout-icon.html>
8. <https://www.iconsdb.com/white-icons/logout-icon.html>
9. <https://www.iconsdb.com/black-icons/map-marker-2-icon.html>
10. <https://www.iconsdb.com/white-icons/map-marker-2-icon.html>
11. <https://www.iconsdb.com/black-icons/list-rich-icon.html>
12. <https://www.iconsdb.com/white-icons/list-rich-icon.html>
13. <https://www.iconsdb.com/black-icons/add-image-icon.html>
14. <https://pngtree.com/freepng/one-star-rating-sign_8424277.html>
15. <https://pngtree.com/freepng/two-star-rating-sign_8423161.html>
16. <https://pngtree.com/freepng/three-star-rating-sign_8411065.html>
17. <https://pngtree.com/freepng/four-star-rating-sign_8436650.html>
18. <https://pngtree.com/freepng/five-star-rating-sign-transparent_8406030.html>

**Logo:**

* <https://album.mediaset.es/cimg/807030/2019/11/04/ef47657facaa7d9d30cd5ae4f3455a8a_965d.jpg>