

Memoria del proyecto

Aplicación web de fórmula 1

Trabajo de Fin de Grado
INGENIERÍA INFORMÁTICA



**VNiVERSiDAD
D SALAMANCA**

Agosto de 2021

Autor:

Daniel Herráez Parro

Tutores:

André Sales Mendes

Juan Francisco de Paz Santana

Gabriel Villarrubia González

Certificado de los tutores

D. André Sales Mendes, D. Juan Francisco De Paz Santana, D. Gabriel Villarubia González, profesores/as del Departamento de Informática y Automática de la Universidad de Salamanca.

CERTIFICAN:

Que el trabajo titulado “Aplicación web de fórmula 1” ha sido realizado por Daniel Herráez Parro, con DNI 70829072C, bajo su dirección.

Y para que conste a todos los efectos oportunos.

En Salamanca, a 4 de septiembre de 2021

D. André Sales Mendes

D. Juan Francisco De Paz Santana

D. Gabriel Villarubia González

Sumario

Este trabajo de fin de grado consiste en el desarrollo de una aplicación web que sea compatible con cualquier dispositivo y accesible desde un navegador web destinada a los aficionados de la Fórmula 1 para seguir más de acerca la competición.

El nombre de la aplicación es F1 APP y pretende ofrecer a los usuarios noticias acerca del mundo de la Fórmula 1, información de cada Gran Premio del campeonato y de los circuitos en los que se realizan las carreras, datos sobre las escuderías y los pilotos que participan en la competición, realizar votaciones cada vez que se celebre un Gran Premio para que cada usuario decida cuál ha sido el mejor piloto de ese fin de semana. Además, con esta aplicación se busca que los usuarios puedan dar a conocer sus opiniones y compartirlas con el resto de los usuarios mediante un foro en el que se puedan tratar diversos temas.

Para el desarrollo de la aplicación se ha empleado JavaScript junto al framework Vue.js, basado en el patrón arquitectónico Modelo-Vista-VistaModelo (MVVM). Para la creación de cada componente de la aplicación se ha empleado Vuetify.js, otro framework altamente compatible con Vue.js que sigue las reglas del diseño Material Design. Para la actualización automática de la clasificación de ha empleado una API para fines no comerciales. Como base de datos, sistema de almacenamiento de contenido multimedia, gestión de la autenticación usuarios y servicio de alojamiento web se han utilizado los servicios de Firebase, herramienta desarrollada por Google al igual que el Material Design.

Todo el proceso de desarrollo del proyecto se ha llevado a cabo siguiendo las pautas del Proceso Unificado.

Palabras clave: JavaScript, Vue.js, Vuetify.js, MVVM, Material Design, API, aplicación web, Fórmula 1.

Summary

This end-of-degree work consists of the development of a web application that is compatible with any device and accesible from a web browser intended for Formula 1 fans to follow more about the competition.

The name of the aplication is F1 APP and it aims to offer users news about the world of Formula 1, information on each Grand Prix of the championship and the circuits where the races are held, data on the teams and drivers that they participate in the competition, take votes each time a Grand Prix is held so that each user decides who has been the best driver of that weekend. In addition, with this apliccaction it is intended that user can make their opinión known and share them with the resto f the users through a fórum in which various topics can be discussed.

For the development of the application, JavaScript has been used together with the Vue.js framework, based on the Model-View-ViewModel (MVVM) architectural pattern. For the creation of each component of the application, Vuetify.js has been used, another framework highly compatible with Vue.js that follows the Material Design rules. An API has been used for the automatic updating of the classification for non-commercial purposes. As a database, multimedia content storage system, user authentication management and web hosting service, the services of Firebase, a tool developed by Google as well as Material Design, have been used.

The entire project development process has been carried out following the guidelines of the Unified Process.

Keywords: JavaScript, Vue.js, Vuetify.js, MVVM, Material Design, API, web application, Formula 1.

Tabla de contenido

1. Introducción.....	1
2. Objetivos del sistema	3
2.1. Objetivos del sistema	3
2.2. Objetivos personales	4
3. Conceptos teóricos	5
3.1. Aplicación web	5
3.2. Frontend	5
3.3. Backend	5
3.4. Material Design	5
3.5. Framework.....	5
3.6. Librería o biblioteca	5
3.7. Firebase	6
3.8. Arquitectura serverless.....	6
3.9. Responsive	6
3.10. API.....	6
3.11. Herramientas CASE.....	7
4. Herramientas utilizadas	9
4.1. Entorno de desarrollo	9
4.1.1. WebStorm.....	9
4.2. Frameworks	9
4.2.1. Vue.js	9
4.2.2. Vuetify.js	9
4.3. Librerías	10
4.3.1. Firebase	10
4.3.2. Vuese	10
4.3.3. JSDoc.....	10
4.3.4. Axios	10
4.4. Lenguajes	10
4.4.1. HTML	10
4.4.2. CSS	11
4.4.3. JavaScript	11
4.4.4. JSON.....	11
4.5. Herramientas de compilación, despliegue y comprobación de código	11
4.5.1. Node.js.....	11
4.5.2. NPM	11
4.5.3. Babel	11
4.5.4. ESLint	12
4.6. Herramientas de control de versiones	12

4.6.1.	GitHub.....	12
4.7.	Herramientas CASE.....	12
4.7.1.	Visual Paradigm	12
4.7.2.	EZEstimate	12
4.7.3.	Microsoft project	12
5.	Aspectos relevantes del desarrollo	13
5.1.	Marco de trabajo.....	13
5.2.	Estimación de costes y esfuerzo	14
5.2.1.	Complejidad de los actores	15
5.2.2.	Complejidad de los casos de uso	15
5.2.3.	Factores de complejidad técnica	16
5.2.4.	Factores de complejidad del entorno.....	17
5.2.5.	Estimación obtenida	18
5.3.	Planificación temporal	19
5.4.	Especificación de requisitos	20
5.4.1.	Objetivos del sistema	20
5.4.2.	Requisitos de información	21
5.4.3.	Actores.....	22
5.4.4.	Requisitos funcionales	22
5.4.5.	Requisitos no funcionales.....	24
5.4.6.	Matriz de rastreabilidad	24
5.5.	Análisis del sistema	25
5.5.1.	Modelo de dominio	25
5.5.2.	Realización de casos de uso.....	26
5.5.3.	Clases de análisis	27
5.5.4.	Paquetes de análisis	28
5.5.5.	Vista de arquitectura del modelo de análisis	28
5.6.	Diseño del sistema.....	29
5.6.1.	Patrones arquitectónicos.....	29
5.6.1.1.	Modelo-Vista-ModeloVista(MVVM)	29
5.6.1.2.	Publicador-Suscriptor	30
5.6.2.	Subsistemas de diseño	31
5.6.3.	Clases de diseño	31
5.6.4.	Vista arquitectónica del modelo de diseño	33
5.6.5.	Realización de casos de uso.....	34
5.6.6.	Diseño de la base de datos.....	34
5.6.7.	Modelo de despliegue	36
6.	Implementación.....	37
6.1.	Configuración de la base de datos	37
6.2.	Desarrollo de la aplicación web.....	38
6.3.	Despliegue de la aplicación	39
6.4.	Documentación técnica	39
7.	Pruebas	41
8.	Funcionalidad de la aplicación.....	43
8.1.	Acceso a la aplicación	43
8.1.1.	Inicio	43

8.1.2.	Registrarse	44
8.1.3.	Recuperar contraseña	45
8.2.	Noticias	46
8.2.1.	Visualización de noticias	46
8.2.2.	Gestión de los comentarios de una noticia	47
8.2.3.	Gestión de las noticias	48
8.3.	Calendario.....	50
8.3.1.	Visualización del calendario.....	50
8.3.2.	Gestión del calendario	51
8.4.	Escuderías	53
8.4.1.	Visualización de las escuderías	53
8.4.2.	Gestión de las escuderías	53
8.5.	Pilotos.....	55
8.5.1.	Visualización de los pilotos.....	55
8.5.2.	Gestión de las pilotos	56
8.6.	Foro	58
8.6.1.	Visualización de los temas	58
8.6.2.	Gestión de los comentarios de un tema	59
8.6.3.	Gestión de los temas	60
8.7.	Votaciones	61
8.7.1.	Visualización de las votaciones.....	61
8.8.	Clasificación	63
8.8.1.	Visualización de las clasificaciones	63
8.8.2.	Actualización de las clasificaciones.....	64
8.9.	Perfil	65
8.9.1.	Visualización del perfil	65
8.9.2.	Configuración del perfil	65
8.9.3.	Cambiar permisos	66
9.	Conclusiones y líneas de trabajo futuras	69
9.1.	Conclusiones	69
9.2.	Líneas de trabajo futuras	70
10.	Bibliografía	71

Índice de figuras

Figura 1: Fases del Proceso Unificado	14
Figura 2: Estimación EZEstimate.....	19
Figura 3: Diagrama de Gantt fase de inicio	20
Figura 4: Diagrama de actores.....	22
Figura 5: Diagrama de casos de uso Gestión de usuarios	23
Figura 6: Modelo de Dominio	25
Figura 7: Diagrama de secuencia - Log in	26
Figura 8: Diagrama de comunicación - Gestión de usuarios	27
Figura 9: Diagrama de paquetes.....	28
Figura 10: Patrón MVVM	29
Figura 11: Patrón Publicador-Suscriptor	30
Figura 12: Diagrama de subsistemas de diseño	31
Figura 13: Diagrama del subsistema Components	32
Figura 14: Diagrama del paquete Login.....	32
Figura 15: Vista arquitectónica del diseño	33
Figura 16: Diagrama de secuencia de diseño - Log in	34
Figura 17: Diagrama de la base de datos.....	35
Figura 18: Diagrama de despliegue	36
Figura 19: Colecciones de Cloud Firestore	37
Figura 20: Inicio de sesión	44
Figura 21: Inicio de sesión – Registrarse	44
Figura 22: Registrarse	45
Figura 23: Recuperar contraseña	46
Figura 24: Noticias	47
Figura 25: Publicar comentario en una noticia	48
Figura 26: Botones para gestionar las noticias.....	48
Figura 27: Publicar noticia	49
Figura 28: Mis noticias	50
Figura 29: Calendario.....	51
Figura 30: Publicar Gran Premio.....	52
Figura 31: Grandes Premios.....	52
Figura 32: Escuderías	53
Figura 33: Publicar escuderías	54
Figura 34: Escuderías	55
Figura 35: Pilotos	56
Figura 36: Publicar piloto.....	57
Figura 37: Pilotos	57
Figura 38: Foro.....	58
Figura 39: Comentarios de un tema	59
Figura 40: Publicar comentario en un tema	59
Figura 41: Publicar tema.....	60
Figura 42: Mis temas	61
Figura 43: Votaciones - votación activa.....	62

Figura 44: Votar	62
Figura 45: Ver votaciones	63
Figura 46: Cerrar o abrir votación	63
Figura 47: Clasificación de escuderías	64
Figura 48: Actualizar clasificación.....	64
Figura 49: Perfil.....	65
Figura 50: Configuración del perfil	66
Figura 51: Configuración perfil - Usuario administrador.....	66
Figura 52: Cambiar permisos	67

Índice de tablas

Tabla 1: Complejidad de los actores (UAW)	15
Tabla 2: Complejidad de casos de uso del paquete Gestión de usuarios	15
Tabla 3: Objetivo - Gestión de usuarios	21
Tabla 4: Caso de uso - Log in	24

1. Introducción

El actual documento contiene la memoria del Trabajo de Fin de Grado “Aplicación web de fórmula 1” en la titulación del Grado en Ingeniería Informática. Ha sido realizado por Daniel Herráez Parro y dirigido por D. André Filipe Sales Mendes, D. Gabriel Villarubia González y D. Juan Francisco De Paz Santana.

Hoy en día, la mayoría de las personas buscan información, noticias e interactúan con otras mediante dispositivos electrónicos a través de diversas aplicaciones que proporcionan esos servicios a los usuarios. Además, lo hacen desde diversos dispositivos ya sean móviles, ordenadores, tablets, etc.

La idea de este proyecto es reunir a una comunidad de personas que les guste la Fórmula 1 y ofrecerles una aplicación donde puedan ver las últimas noticias sobre este deporte, información acerca de la competición, que tengan la posibilidad de compartir sus opiniones con otros seguidores de la Fórmula 1 y crear un sitio donde puedan debatir sobre el mundo del motor.

A lo largo de este documento se describen los aspectos más relevantes de la realización del proyecto que se divide en los siguientes apartados:

- **Objetivos del sistema:** Se exponen los objetivos que se quieren cumplir con el desarrollo del proyecto.
- **Conceptos teóricos:** Se muestran conceptos teóricos relacionados con el proyecto que son útiles para su comprensión.
- **Herramientas utilizadas:** Se exponen las técnicas y herramientas empleadas para el desarrollo del proyecto.
- **Aspectos relevantes del desarrollo:** Se describen los aspectos más relevantes del desarrollo del proyecto.
- **Implementación:** Se exponen aspectos relacionados con la implementación de la aplicación final.
- **Pruebas:** Se comentan las pruebas que se han realizado sobre los subsistemas y componentes de la aplicación.
- **Funcionalidad de la aplicación:** Se explica y muestra las diferentes funcionalidades de la aplicación.
- **Conclusiones y líneas futuras de trabajo:** Se comentan las conclusiones resultantes de la realización del proyecto y posibles mejoras a implementar en un futuro.

Además de este documento se proporciona el código de la aplicación y la documentación que se encuentra distribuida en los siguientes anexos:

Aplicación web de fórmula 1

- **Anexo I – Plan de proyecto:** Se realiza la estimación del esfuerzo y la planificación temporal de las tareas del proyecto.
- **Anexo II – Especificación de requisitos software:** Se realiza la especificación de los requisitos software que debe cumplir la aplicación.
- **Anexo III – Análisis de requisitos:** Se realiza un análisis de los requisitos software, su refinamiento y un primer paso para la estructura de la aplicación.
- **Anexo IV – Diseño del sistema software:** Se documenta el modelo de diseño del sistema.
- **Anexo V – Documentación técnica:** Se expone la documentación técnica del código de la aplicación con el objetivo de facilitar la comprensión del código por cualquier desarrollador.
- **Anexo VI – Manual de usuario:** Sirve de guía para cualquier usuario que acceda al sistema.

2. Objetivos del sistema

En este apartado se exponen los objetivos principales que debe tener el sistema planteado para satisfacer los requisitos establecidos. Además, de los objetivos personales que se quieren lograr con el desarrollo de este proyecto.

2.1. Objetivos del sistema

El objetivo principal es desarrollar y diseñar una aplicación web destinada a mostrar información sobre Fórmula 1 a cualquier usuario que acceda a ella y los objetivos que debe satisfacer son los siguientes:

- **Gestión de usuarios:** el sistema deberá permitir crear cuentas de usuarios a aquellos que no estén registrados, modificar sus datos en cualquier momento y recuperar su contraseña. Además, controlará el acceso de estos usuarios a las funcionalidades de la aplicación en función de sus privilegios dentro del sistema.
- **Almacenamiento y visualización de información:** el sistema deberá almacenar toda la información necesaria de las noticias, el calendario, las escuderías, los pilotos, los temas, los comentarios, las votaciones y sus resultados para poder mostrarla a cualquier usuario que acceda a la aplicación.
- **Gestión de noticias:** el sistema deberá permitir a todos los usuarios visualizar las noticias, crear, editar o eliminar comentarios sobre cualquiera de ellas, y a los usuarios con privilegios de administrador crear, editar y eliminar noticias en el sistema.
- **Gestión del foro:** el sistema deberá permitir a todos los usuarios publicar temas en el foro, y editar o eliminar aquellos hayan sido creados por ellos. Además, permitirá a los usuarios hacer comentarios en todos los temas y editarlos o eliminarlos.
- **Gestión de las votaciones:** el sistema deberá permitir a los usuarios votar a un piloto en cada votación. El sistema calculará los resultados de las votaciones y guardará un histórico con los tres pilotos más votados en cada votación para que los usuarios puedan ver los porcentajes de los tres pilotos más votados en cada votación.
- **Gestión de la clasificación:** el sistema deberá permitir actualizar la clasificación de los pilotos y las escuderías cuando un usuario con privilegios lo solicite. Además, de mostrar ambas clasificaciones para cualquier usuario que acceda a la aplicación.
- **Gestión del calendario:** el sistema deberá permitir a todos los usuarios visualizar todas carreras del campeonato y a los usuarios con privilegios de administrador crear, editar y eliminar un Gran Premio en el sistema.
- **Gestión de las escuderías:** el sistema deberá permitir a todos los usuarios

Aplicación web de fórmula 1

visualizar todas las escuderías que participan y a los usuarios con privilegios de administrador crear, editar y eliminar escuderías en el sistema.

- Gestión de los pilotos: el sistema deberá permitir a todos los usuarios visualizar todos los pilotos que participan y a los usuarios con privilegios de administrador crear, editar y eliminar pilotos en el sistema.

Además, el sistema deberá permitir el acceso de forma simultánea a todos los usuarios que deseen acceder a él, ser sencillo y fácil de utilizar para cualquier usuario, para ello debe proporcionar una interfaz intuitiva, también deberá poder ser utilizado desde cualquier dispositivo, ya sea un móvil, una tablet o un ordenador, garantizando el correcto funcionamiento, siendo lo suficientemente fiable para que no se caiga muchas veces y garantizar la integridad de los datos. Por último, deberá funcionar de manera eficiente en cualquier dispositivo y cumplir con unos tiempos de respuesta oportunos a las peticiones que realicen los usuarios para lograr una satisfacción de uso.

2.2. Objetivos personales

Por otro lado, dentro de los objetivos personales de este proyecto cabe destacar el desarrollo de una aplicación informática completa, desde el comienzo de una idea hasta obtener el sistema final y la realización de las pruebas. De este modo me ha permitido aplicar los conocimientos adquiridos durante la formación académica estos últimos cuatro años.

Además, el hecho de construir un sistema completo supone tener conocimientos sobre programación orientada a objetos e ingeniería software y entender los problemas que solucionan las herramientas proporcionadas por estas disciplinas.

Por último, el hecho de que sea la primera vez que desarrollo una aplicación web y empleo de herramientas nuevas como Vue.js, Vuetify.js, Firebase y JavaScript; supone un reto añadido al desarrollo del proyecto.

3. Conceptos teóricos

3.1. Aplicación web

Una aplicación web es una herramienta que pueden utilizar los usuarios a través de un navegador web accediendo a un servidor web mediante internet. Básicamente son programas interpretados por los navegadores web que se encargan de su ejecución.

Este tipo de aplicaciones tiene las ventajas de funcionar como clientes ligeros, independientemente del sistema operativo y la facilidad para realizar actualizaciones y mantener aplicaciones web si necesidad de instalar el software a todos los usuarios que la utilicen. Además, de poder ser visualizadas en cualquier dispositivo sin importar sus medidas.

3.2. Frontend

Es un término que hace referencia a la parte de una aplicación con la que interactúa un usuario.

3.3. Backend

Representa el almacenamiento y tratamiento de los datos, la parte que no interactúa de forma directa con el usuario.

3.4. Material Design

Es una normativa de diseño para la creación de interfaces de usuario. Desarrollado por Google con el objetivo de construir interfaces intuitivas, sencillas y visualmente agradables para los usuarios.

3.5. Framework

Es un conjunto estandarizado de conceptos, criterios, prácticas, módulos empleados en el desarrollo de un proyecto software con el objetivo facilitar el proceso de desarrollo, debido a que sirve de referencia para enfrentar y resolver nuevos problemas similares.

3.6. Librería o biblioteca

Es un conjunto de implementaciones funcionales realizadas en cualquier lenguaje de programación, empleadas para extender la funcionalidad de una aplicación.

Aplicación web de fórmula 1

3.7. Firebase

Firebase es una plataforma en la nube destinada para el desarrollo de aplicaciones web y móvil, que presta almacenamiento y sirve como base de datos, propiedad de Google. Ofrece diversos servicios de backend y para el desarrollo de este proyecto se han empleado los siguientes:

- Authentication: es un servicio de backend que permite gestionar todos los procesos de autenticación de los usuarios en una aplicación.
- Cloud Firestore: es una base de datos NoSQL flexible, escalable y en la nube, cuyo fin es almacenar y sincronizar datos. Este servicio permite el acceso a los datos mediante suscripciones a cambios que se produzcan en las colecciones o los documentos almacenados.
- Cloud Storage: es un servicio que permite almacenar audios, fotos, videos de manera sencilla y segura.
- Hosting: es un servicio que permite hospedar aplicaciones web, entregando contenidos dinámicos y estáticos. Además, incluye SSL sin necesidad de configuración para que el contenido se publique de manera segura.

3.8. Arquitectura serverless

Una arquitectura serverless (sin servidor), significa que no hay una administración de los servidores por los desarrolladores, sino que los servidores son administrados por proveedores como Amazon, Google, etc.

Una variante de esta arquitectura es conocida como BaaS (Backend as a Service) donde el proveedor presta una serie de servicios como la autenticación de usuarios, base de datos, almacenamiento, hosting, etc. Firebase es un claro ejemplo de este tipo de arquitectura.

Las aplicaciones que siguen este tipo de arquitectura tienen ventajas como la facilidad en el despliegue y escalado [1].

3.9. Responsive

El diseño web responsive o adaptativo es una técnica que pretende conseguir la visualización correcta una página en diferentes dispositivos. Mediante la colocación de los elementos y su redimensionado se consigue una correcta visualización y una mejor experiencia de uso.

3.10. API

Una interfaz de programación de aplicaciones (API) es un conjunto de funciones y métodos que proporciona una biblioteca para ser utilizada por otro software como una

Aplicación web de fórmula 1

capa de abstracción.

3.11. Herramientas CASE

Es un conjunto de aplicaciones informáticas destinadas a facilitar el desarrollo de software reduciendo el costo en tiempo y dinero.

4. Herramientas utilizadas

Es un conjunto de aplicaciones informáticas destinadas a facilitar el desarrollo de software reduciendo el costo en tiempo y dinero.

4.1. Entorno de desarrollo

4.1.1. WebStorm

Es un entorno de desarrollo integrado para JavaScript y tecnologías relacionadas como HTML y CSS. Desarrollado por JetBrains y centrado en el desarrollo web y es compatible con Vue.js, React, Node.js, ect.

Es un editor inteligente que incluye funcionalidades de autocompletado de código, detección de errores, sugerencias para la corrección de estos y permite realizar refactorizaciones de código de manera segura [2].

Las razones de la utilización de este entorno de desarrollo han sido su compatibilidad con Vue.js y su licencia libre de acceso.

4.2. Frameworks

4.2.1. Vue.js

Es un framework de código abierto de JavaScript creado por Evan You, que permite la construcción de interfaces de usuario y aplicaciones de una sola página. Este framework emplea un alto desacoplamiento permitiendo a los desarrolladores crear progresivamente interfaces de usuario mediante el uso de componentes, que son elementos que contienen código HTML, JavaScript y CSS. Además, sigue el patrón arquitectónico MVVM (Model-View-ViewModel) [3].

La elección del framework ha sido por la gran cantidad de recursos y documentación existente.

4.2.2. Vuetify.js

Es un framework progresivo de componentes para Vue.js, permite acelerar el desarrollo de interfaces de usuario proporcionando elementos como botones, tablas, formularios, etc. siguiendo la estética del Material Design [4].

Se ha empleado este framework debido a su alta compatibilidad con Vue.js.

Aplicación web de fórmula 1

4.3. Librerías

4.3.1. Firebase

Librería para Vue.js que permite el acceso a los diferentes servicios proporcionados por Firebase [5].

Los motivos de su utilización han sido por las funcionalidades que ofrece y su facilidad de uso.

4.3.2. Vuese

Es una librería que facilita herramientas para la creación de la documentación del proyecto de manera rápida y sencilla de componentes Vue.js. Vuese genera una documentación en formato HTML mediante comentarios específicos en cada componente [6].

La elección de esta librería es por estar desarrollada para realizar documentación de componentes Vue.js.

4.3.3. JSDoc

Es una librería que permite la documentar de forma rápida y sencilla ficheros JavaScript. Al igual que Vuese, JSDoc también genera archivos en formato HTML con la información del código [7].

El motivo de su utilización ha sido porque permite generar documentación de archivos JavaScript.

4.3.4. Axios

Es una librería JavaScript que puede ejecutarse en el navegador y facilita la realización de operaciones como cliente HTTP. En este proyecto se emplea para realizar peticiones GET a la API que contiene los datos de la clasificación de pilotos y escuderías [8]. La API empleada para obtener los datos sobre la clasificación:

<https://ergast.com/mrd/>

4.4. Lenguajes

4.4.1. HTML

El lenguaje de marcas de hipertexto (HyperText Markup Lenguaje) es el componente más básico de la web, define la estructura del contenido e identificada cada uno de sus elementos [9].

Aplicación web de fórmula 1

4.4.2. CSS

Las hojas de estilos en cascada (Cascading Style Sheets) es un lenguaje de estilos empleado para describir la presentación de documentos HTML y cada uno de sus elementos [10].

4.4.3. JavaScript

Es lenguaje de programación interpretado, orientado a objetos, interpretativo y declarativo, se emplea para añadir una secuencia de comando en páginas web y así dotarlas de una cierta funcionalidad [11].

4.4.4. JSON

La notación de objetos de JavaScript (JavaScript Object Notation) es un formato ligero de intercambio de datos. Tiene las características de que es simple de leer y escribir por humanos; y simple de interpretar y generar por máquinas.

En la aplicación es empleado por la API para devolver la información de clasificación y también es utilizado por Firabase para almacenar datos [12].

4.5. Herramientas de compilación, despliegue y comprobación de código

4.5.1. Node.js

Es un entorno de ejecución de JavaScript, incluye todo lo necesario para ejecutar un programa escrito en JavaScript [13]. Fue creado por los desarrolladores de JavaScript. Tanto JavaScript como Node.js se ejecutan en el motor de tiempo de ejecución JavaScript V8, que se encarga de convertir código JavaScript en código máquina.

4.5.2. NPM

Npm es el gestor de paquetes desarrollado mediante el lenguaje JavaScript para Node.js. Se ejecuta desde la línea de comando y permite instalar bibliotecas y plugins que se encuentren en el repositorio, de este modo permite agregar dependencias de manera fácil y rápida [14].

4.5.3. Babel

Es un compilador que permite codificar el código JavaScript más moderno en código JavaScript que cualquier navegador o versión de Node.js pueda entender, funciona mediante plugins para que podamos indicar que cosas queremos que transforme [15].

Aplicación web de fórmula 1

4.5.4. ESLint

Es un analizador de código que permite detectar errores que podrían provocar problemas durante la compilación o generar futuros bugs en el programa. Además, permite establecer un estilo de código homogéneo para el desarrollo y definir reglas para que se cumplan y se apliquen en el código [16].

4.6. Herramientas de control de versiones

4.6.1. GitHub

Es un servicio de alojamientos de repositorios de Git que permite desarrollar proyectos software en los que colaboren varias personas y emplea Git como sistema de control de versiones, este repositorio es ampliamente utilizado para desarrollar proyectos y permitir que cualquier persona pueda colaborar en ellos [17].

4.7. Herramientas CASE

4.7.1. Visual Paradigm

Es una herramienta que permite diseñar diagramas UML de manera sencilla, establecer relaciones entre componentes y clases entre distintos diagramas, permite elaborar el ciclo de vida completo del desarrollo software, desde la elicitación de requisitos, análisis, diseño, construcción y despliegue [18].

4.7.2. EZEstimate

Es una herramienta que permite realizar una estimación del esfuerzo de desarrollo de un proyecto software a partir de los puntos de caso de uso en función de la complejidad de estos, de los actores involucrados en el sistema y de los factores de complejidad técnica y del entorno.

4.7.3. Microsoft project

Es una herramienta para la gestión de proyectos creada por Microsoft que permite organizar y llevar a cabo un seguimiento de las tareas de un proyecto. Mediante esta herramienta se pueden asignar las tareas a diferentes recursos, seguir el progreso, administrar presupuestos, gestionar las cargas de trabajo de los recursos y maximizar su rendimiento [19].

5. Aspectos relevantes del desarrollo

A continuación, se van a exponer los aspectos más relevantes de las diferentes fases del desarrollo de la aplicación.

5.1. Marco de trabajo

Como marco de trabajo se ha empleado el Proceso Unificado de Desarrollo Software. Caracterizado por estar dirigido por los casos de uso, centrado en la arquitectura y por ser iterativo e incremental.

El Proceso Unificado se divide en fases, la finalización de cada fase es un hito principal. Además, cada una de las fases se divide en iteraciones formadas por diversas tareas y tienen como resultados hitos. A su vez, las tareas se clasifican en disciplinas: Modelado de negocio, Requisitos, Análisis, Diseño, Implementación, Pruebas, Configuración y Manejo. Las 4 fases del Proceso Unificado son:

- **Inicio:** Realización del modelado de negocio y definición de magnitud del proyecto.
- **Elaboración:** Especificación de los aspectos funcionales del sistema.
- **Construcción:** Implementación del sistema y realización de pruebas.
- **Transición:** Realización de pruebas y tareas de mantenimiento una vez se obtiene una versión final del sistema.

En la *Figura 1* se muestra el flujo de trabajo con las distintas iteraciones y las disciplinas que tienen lugar en las diferentes fases:

Aplicación web de fórmula 1

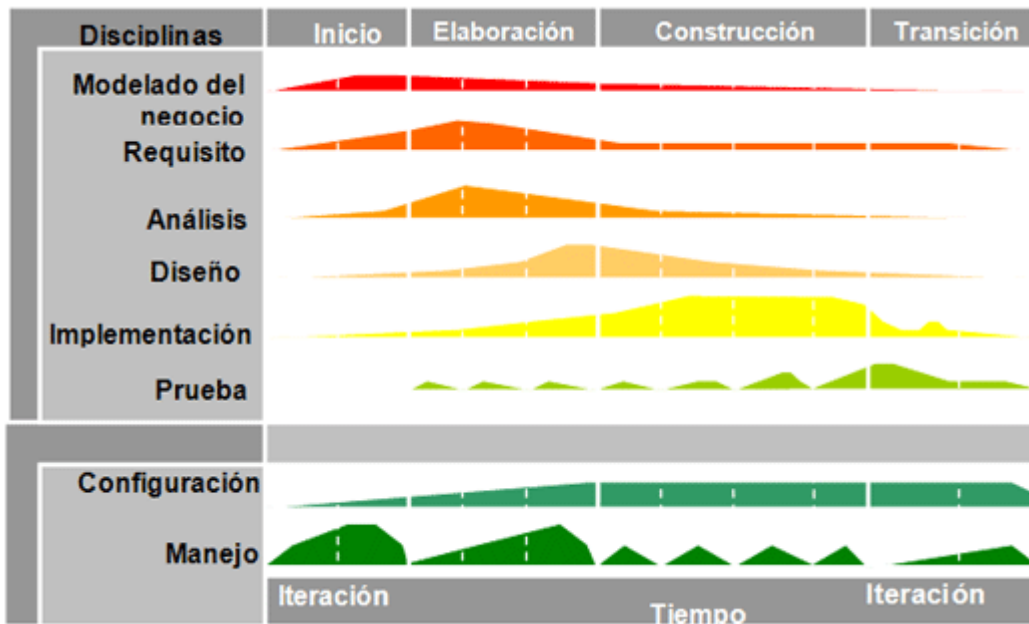


Figura 1: Fases del Proceso Unificado

El hecho de haber seleccionado este marco de trabajo se debe a que ya se ha trabajado con él durante algunas asignaturas del grado.

5.2. Estimación de costes y esfuerzo

Mediante la estimación de coste y esfuerzo se puede realizar una estimación de la dimensión del proyecto y el esfuerzo necesario para poder desarrollarlo. Para ello se ha empleado la métrica **UCP** (Use Case Points), basada en la funcionalidad representada en forma de casos de uso [20].

Se requiere del cálculo de las siguientes variables:

- **UUCP** (Unadjusted Use Case Points): puntos de casos de uso desajustados. Su valor se obtiene de la suma de las siguientes variables:
 - **UUCW** (Unadjusted Use Case Weight): peso de los casos de uso desajustado.
 - **UAW** (Unadjusted Actor Weight): peso de los actores desajustado.
- **TCF** (Technical Complexity Factor): factor de complejidad técnica.
- **ECF** (Environment Complexity Factor): factor de complejidad del entorno.

La aplicación de estas variables en la siguiente ecuación permite calcular los UCP:

$$UCP = UUCP \times TCF \times ECF$$

La estimación del esfuerzo en número de horas persona se obtiene mediante la

Aplicación web de fórmula 1

siguiente ecuación:

$$Esfuerzo = UCP \times F$$

5.2.1. Complejidad de los actores

Para el sistema se han establecido 3 actores como se muestran en la *Tabla 1*, con una complejidad asociada a cada actor.

Actor	Descripción	Complejidad	Peso
Usuario administrador	Interactúa con el sistema mediante una interfaz gráfica.	Complejo	3
Usuario no identificado	Interactúa con el sistema mediante una interfaz gráfica.	Complejo	3
Usuario identificado	Interactúa con el sistema mediante una interfaz gráfica.	Complejo	3

Tabla 1: Complejidad de los actores (UAW)

5.2.2. Complejidad de los casos de uso

A continuación, en la *Tabla 2* se muestran algunos casos de uso del paquete de Gestión de usuarios, el número de transacciones contabilizadas, la complejidad y el peso de cada uno de ellos:

Caso de uso	Transacciones	Complejidad	Peso
Log in	2	Simple	5
Registro	2	Simple	5
Recuperar contraseña	3	Simple	5
Log out	1	Simple	5
Ver perfil	1	Simple	5
Modificar datos	2	Simple	5
Cambiar contraseña	2	Simple	5
Cambiar correo	2	Simple	5
Modificar privilegios	2	Simple	5

Tabla 2: Complejidad de casos de uso del paquete Gestión de usuarios

Aplicación web de fórmula 1

5.2.3. Factores de complejidad técnica

Mediante estos 13 factores de complejidad técnica se calcula el TCF aplicando la siguiente fórmula, donde W_i es el peso del factor i -ésimo y F_i es la complejidad percibida del i -ésimo factor que puede ir desde 0 hasta 5. C_1 y C_2 son constantes.

$$TCF = C_1 + C_2 \sum_{i=1}^{13} W_i F_i$$

Para el caso este sistema la complejidad asignada a cada factor son las siguientes:

- **T1 - Sistema distribuido: 2.**
Se trata de una aplicación que se desplegará en un único servidor y puede utilizarse en varios dispositivos a través de un navegador.
- **T2 - Rendimiento: 2.**
La aplicación debe de procesar las peticiones de los usuarios con cierta rapidez para una buena experiencia de uso, aunque se puede permitir un cierto retardo.
- **T3 - Eficiencia del usuario final: 3.**
La aplicación debe permitir a los usuarios realizar todas las tareas de manera rápida y sencilla, buscando que el usuario pueda utilizar cualquier funcionalidad en el menor tiempo posible.
- **T4 - Complejidad de procesamiento interno: 1.**
El proceso de ejecución interno no requiere de unas transacciones complejas, ya que los cálculos y la información que maneja son sencillos.
- **T5 - Reusabilidad: 2.**
Este aspecto no es muy necesario enfocarse en él, aunque si es importante hacer un desarrollo que permita futuras extensiones.
- **T6 - Facilidad de instalación: 1.**
La aplicación será fácilmente accesible a través del navegador, por lo que no requiere de un proceso de instalación.
- **T7 - Facilidad de uso: 3.**
Los usuarios que pueden acceder a la aplicación pueden ser de diferentes edades, por lo que se intenta que la distribución de elementos sea la mejor para facilitar la utilización a cualquier usuario.
- **T8 - Portabilidad: 3.**

Aplicación web de fórmula 1

Como se trata de una aplicación web hay que adaptarla a las distintas resoluciones de pantalla de los dispositivos que puedan conectarse.

- **T9 - Facilidad de cambio: 2.**
Debido a que los requisitos iniciales son claros y concisos no se prevé ningún cambio en ellos, aunque la aplicación debe permitir añadir nuevas funcionalidades con facilidad.
- **T10 - Concurrencia: 3.**
El sistema debe permitir el acceso concurrente a varios usuarios.
- **T11 - Características especiales de seguridad: 2.**
No es necesario incluir ningún aspecto especial de seguridad, únicamente la protección de la base de datos, donde se alojarán todos los datos referentes a los usuarios y asegurar que solo pueden acceder usuarios registrados en la aplicación.
- **T12 - Acceso directo a terceras partes: 0.**
No se requiere de interfaces a terceras partes.
- **T13 - Se requiere de entrenamiento especial del usuario: 1.**
La aplicación será muy intuitiva para los usuarios, por lo que no se requiere de ninguna documentación que sirva de guía para los usuarios.

5.2.4. Factores de complejidad del entorno

Mediante estos 8 factores de complejidad del entorno se calcula el ECF aplicando la siguiente fórmula, donde W_i es el peso del factor i -ésimo y F_i es la complejidad percibida del i -ésimo factor que puede ir desde 0 hasta 5. C_1 y C_2 son constantes.

$$ECF = C_1 + C_2 \sum_{i=1}^8 W_i F_i$$

Para el caso este sistema la complejidad asignada a cada factor son las siguientes:

- **E1 - Familiaridad con UML: 2.**
Durante algunas asignaturas del grado se ha trabajado con el proceso unificado, pero no se ha realizado ningún proyecto completo.
- **E2 - Trabajadores a tiempo parcial: 3.**
El proyecto es realizado por una única persona que lo compagina con asignaturas del último curso, prácticas y trabajo en una empresa.

Aplicación web de fórmula 1

- **E3 - Capacidad de los analistas: 2.**
Se cuenta con poca experiencia en actividades de análisis en el desarrollo de un proyecto software.
- **E4 - Experiencia en la aplicación: 2.**
Se tiene poca experiencia con los lenguajes y herramientas que se van a emplear.
- **E5 - Experiencia en la orientación a objetos: 3.**
Se ha adquirido cierta experiencia en la utilización de varios lenguajes de programación orientados a objetos durante el grado.
- **E6 - Motivación: 5.**
La motivación es alta porque supone la finalización del grado y la temática del proyecto es una afición del desarrollador.
- **E7 - Dificultad del lenguaje: 2.**
Los lenguajes de programación que se van a emplear son de alto nivel y no son excesivamente complicados, aunque no se cuenta con mucha experiencia en ellos.
- **E8 - Estabilidad de los requisitos: 4.**
Los requisitos están bien definidos y es difícil que sufran alguna modificación durante el desarrollo, aunque se puede dar lugar a pequeñas modificaciones de mejora.

5.2.5. Estimación obtenida

Mediante la herramienta EZEstimate se realizan los cálculos y se obtiene una estimación del coste y esfuerzo de desarrollo, como se puede ver en la *Figura 2*.

Aplicación web de fórmula 1

Estimation Summary	
UAW	9
UUCW	225
UUPC = UAW + UUCW	234
TFactor	30
EFactor	16
TCF = 0.6 + (.01*TFactor)	0,9
EF = 1.4 + (-0.03*EFactor)	0,92
UCP = UUCP*TCT*EF	193,752
Total Effort@ <input type="text" value="5"/>	Hrs/UCP 968,76

Figura 2: Estimación EZEstimate

El resultado que proporciona EZEstimate son 968,76 horas aproximadamente para el desarrollo del proyecto. Si se emplean una media de 6 horas por día, el proyecto tiene una estimación de duración de 161,46 días.

5.3. Planificación temporal

En la planificación temporal se identifican las tareas que deben realizarse para llevar a cabo el proyecto, gracias a ello se podrá obtener una idea del tiempo necesario para la realización del proyecto. Para el desarrollo de la planificación temporal se ha empleado la herramienta Microsoft Project [19].

Para comenzar se ajusta la fecha de inicio del proyecto y el horario con los días del calendario que se van a dedicar para el proyecto, está previsto que se trabajen todos los días del calendario incluidos fines de semana y festivos. Este calendario puede encontrarse en el **Anexo I – Plan de proyecto**. La fecha de comienzo es el 15 de marzo de 2021.

A continuación, en la *Figura 3* se muestran las tareas de la fase de inicio del proyecto, las dependencias entre ellas, sus duraciones y el camino crítico mediante el diagrama de Gantt:

Aplicación web de fórmula 1

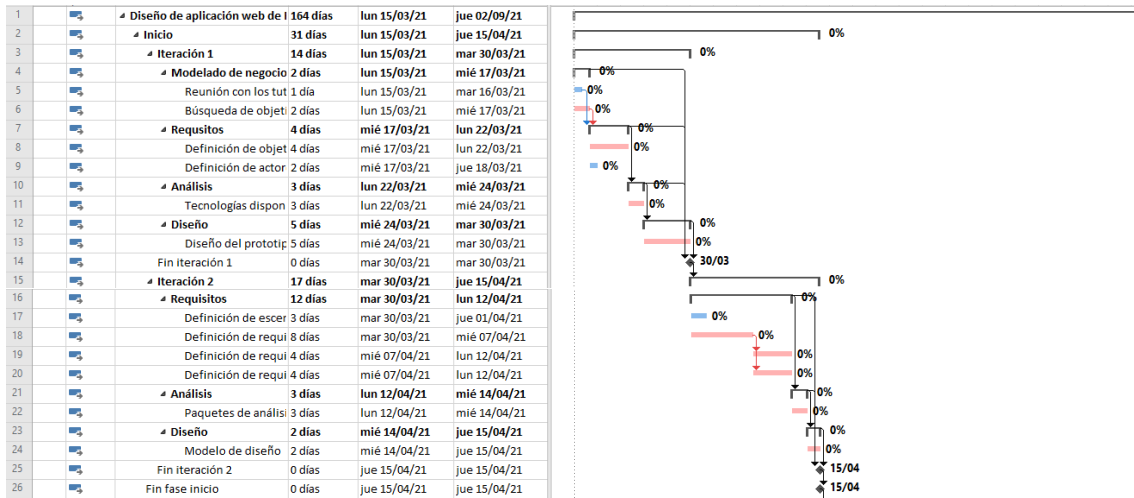


Figura 3: Diagrama de Gantt fase de inicio

5.4. Especificación de requisitos

En este apartado se recoge a especificación de requisitos software del sistema a desarrollar. Para ello se ha empleado la metodología de Durán y Bernárdez como guía para su realización, que proporciona una estructura para la elicitación de requisitos software.

A continuación, se muestra un resumen de la aplicación de esta metodología en sobre la aplicación. La documentación completa de este apartado se encuentra en el documento: **Anexo II – Especificación de requisitos software**.

5.4.1. Objetivos del sistema

Los objetivos para cumplir por el sistema son:

- Gestión de usuarios
- Almacenamiento y visualización de información
- Gestión de noticias
- Gestión del foro
- Gestión de las votaciones
- Gestión de la clasificación
- Gestión del calendario
- Gestión de los equipos
- Gestión de los pilotos

Para describir cada uno de estos objetivos se han empleado tablas similares a la *Tabla 3* que describe la Gestión de usuarios:

Aplicación web de fórmula 1

OBJ-0001	Gestión de usuarios
Versión	1.0 (26/05/2021)
Autores	Herráez Parro, Daniel
Fuentes	Sales Mendes, André Filipe De Paz Santana, Juan Francisco Villarrubia González, Gabriel
Descripción	El sistema deberá <i>permitir crear cuentas de usuarios a aquellos que no estén registrados, modificar sus datos en cualquier momento y recuperar su contraseña. Además, controlará el acceso de estos usuarios a las funcionalidades de la aplicación en función de sus privilegios dentro del sistema.</i>
Subobjetivos	Ninguno
Importancia	Vital
Urgencia	Inmediatamente
Estado	Validado
Estabilidad	Alta
Comentarios	Ninguno

Tabla 3: Objetivo - Gestión de usuarios

5.4.2. Requisitos de información

Estos indican la información a almacenar por el sistema:

- Información sobre usuarios
- Información sobre temas
- Información sobre comentarios
- Información sobre escuderías
- Información sobre pilotos
- Información sobre noticias
- Información sobre Gran Premio
- Información sobre votos
- Información sobre votaciones

Para describir cada uno de estos requisitos se han empleado tablas que se puede ver en el **Anexo II – Especificación de requisitos software**.

Aplicación web de fórmula 1

5.4.3. Actores

Los actores que intervienen en el sistema, representado a usuarios que interactúan con el sistema, y su jerarquía:

- **Usuario identificado:** usuario que accede a la aplicación una vez que se ha autenticado con sus credenciales.
- **Usuario no identificado:** usuario que todavía no ha accedido a la aplicación y no se ha autenticado en el sistema.
- **Administrador:** usuario registrado en el sistema, que accede a la aplicación una vez que se ha autenticado con sus credenciales y posee ciertos privilegios dentro del sistema.

En la *Figura 4* se puede ver el diagrama de los actores involucrados en el sistema:

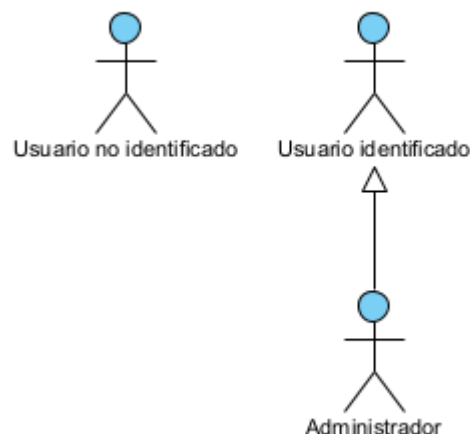


Figura 4: Diagrama de actores

5.4.4. Requisitos funcionales

Estos especifican una funcionalidad o los servicios que debe proveer la aplicación, se recogen mediante los casos de uso, que se agrupan en paquetes según su funcionalidad.

Los diagramas empleados para indicar los casos de uso de un paquete, las relaciones entre ellos y los actores son similares al mostrado en la *Figura 5* sobre el paquete de Gestión de usuarios:

Aplicación web de fórmula 1

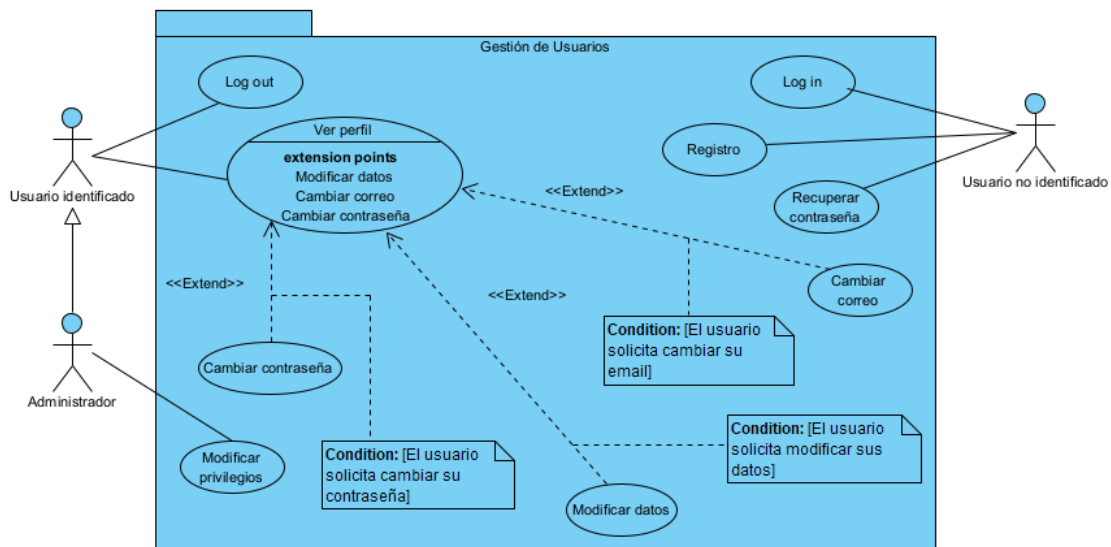


Figura 5: Diagrama de casos de uso Gestión de usuarios

Para describir cada uno de los casos de uso se han empleado tablas similares a la Tabla 4 que describe el caso de uso Log in:

UC-0001	Log in	
Versión	1.0 (27/05/2021)	
Autores	Herráez Parro, Daniel	
Fuentes	Sales Mendes, André Filipe De Paz Santana, Juan Francisco Villarrubia González, Gabriel	
Dependencias	<ul style="list-style-type: none"> ¡Error! No se encuentra el origen de la referencia. 	
Descripción	El sistema debe comportarse tal como se describe en el siguiente caso de uso cuando <i>el Usuario no identificado solicita iniciar sesión en el sistema.</i>	
Precondición	El Usuario no está identificado en el sistema.	
Secuencia normal	Paso	Acción
	1	El sistema solicita al usuario su correo electrónico y contraseña.
	2	El actor ¡Error! No se encuentra el origen de la referencia. introduce los datos solicitados, y solicita el acceso.
	3	El sistema recibe los datos y comprueba que sean correctos, proporcionando el acceso al sistema.

Aplicación web de fórmula 1

Postcondición	El Usuario no identificado pasa al estado identificado en el sistema.	
Excepciones	Paso	Acción
	3	Si algún campo no se ha especificado, no es válido o el correo electrónico y contraseña no coinciden el sistema informa al usuario del error.
Rendimiento	Paso	Tiempo máximo
	-	-
Frecuencia	3 veces por día(s)	
Importancia	Vital	
Urgencia	Inmediata	
Estado	Valido	
Estabilidad	Alta	
Comentarios	Ninguno	

Tabla 4: Caso de uso - Log in

5.4.5. Requisitos no funcionales

Estos no describen información que debe almacenar el sistema, ni funciones que debe realizar, sino características de funcionamiento que debe poseer el sistema. Sirven para especificar las restricciones o condiciones que se tienen que cumplir en el diseño y en la implementación de mismo:

- Concurrencia
- Compatibilidad
- Usabilidad
- Fiabilidad
- Rendimiento
- Privacidad
- Seguridad

Para describir cada uno de estos requisitos se han empleado tablas que se puede ver en el **Anexo II – Especificación de requisitos software**.

5.4.6. Matriz de rastreabilidad

La matriz de rastreabilidad permite conocer las relaciones entre los objetivos del sistema con cada uno de los casos de usos definidos en el sistema. Se encuentra en el documento: **Anexo II – Especificación de requisitos software**

Aplicación web de fórmula 1

5.5. Análisis del sistema

En este apartado se recoge el análisis de los requisitos software del sistema a desarrollar. Con ello se consigue tener una primera representación de la arquitectura del sistema que será refinada en el diseño

La documentación completa de este apartado se encuentra en el documento: **Anexo III – Análisis de los requisitos.**

5.5.1. Modelo de dominio

El modelo de dominio es una representación de las clases conceptuales significativas del dominio del problema, en él se recoge las necesidades de almacenamiento y gestión de la información del sistema.

En la *Figura 6* se muestra el diagrama de clases que representa el modelo de dominio:

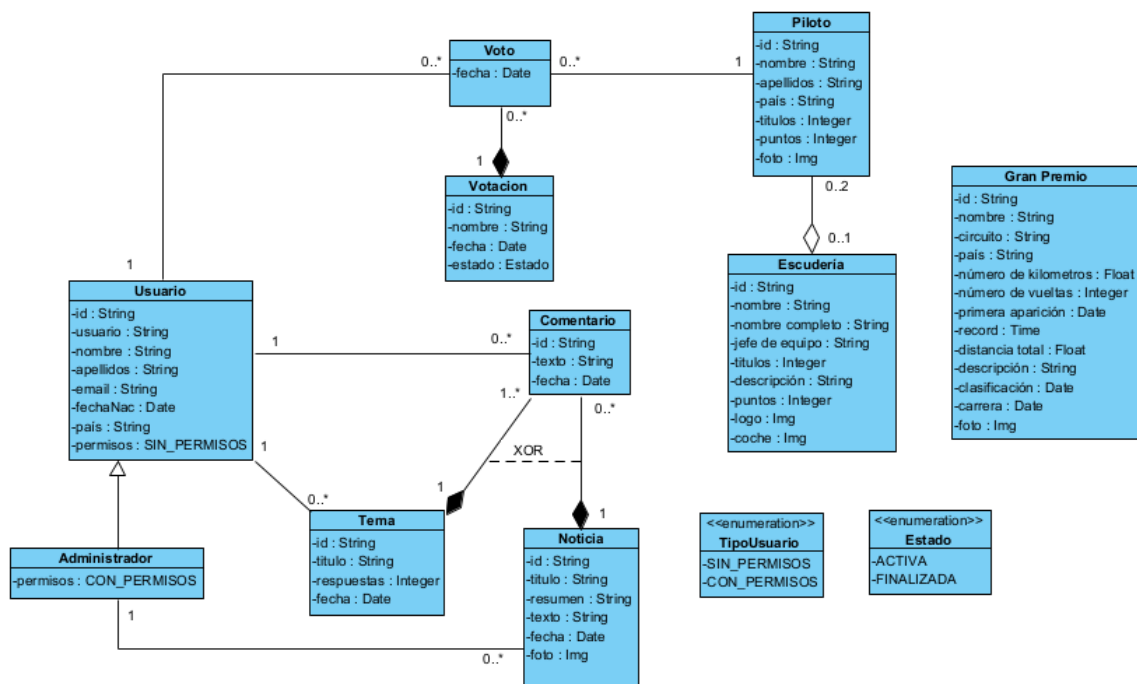


Figura 6: Modelo de Dominio

Aplicación web de fórmula 1

5.5.2. Realización de casos de uso

La realización de casos de uso se describe mediante diagramas de secuencia que describen el intercambio de mensajes entre objetos. En la *Figura 7* se muestra un diagrama de secuencia del caso de uso Log in:

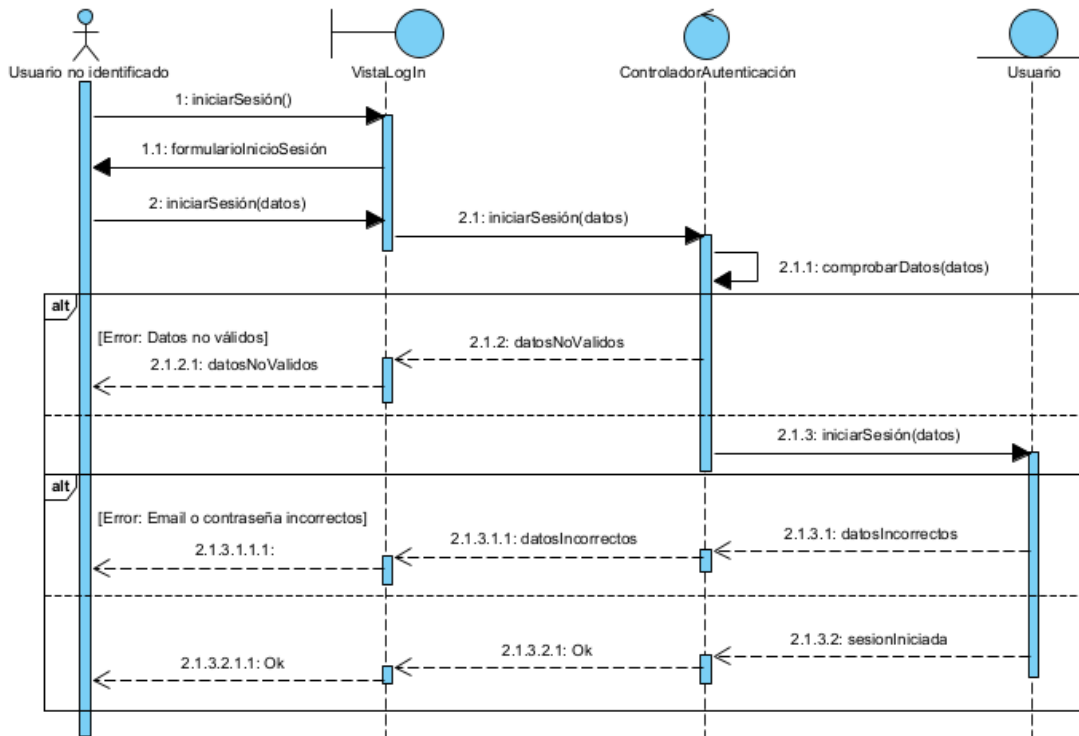


Figura 7: Diagrama de secuencia - Log in

Aplicación web de fórmula 1

5.5.3. Clases de análisis

En este apartado se muestran los diagramas de comunicación de las clases de análisis que se han obtenido durante el desarrollo de los diagramas de secuencia de los casos de uso. En la *Figura 8* se muestra el diagrama de comunicación del paquete Gestión de usuarios:

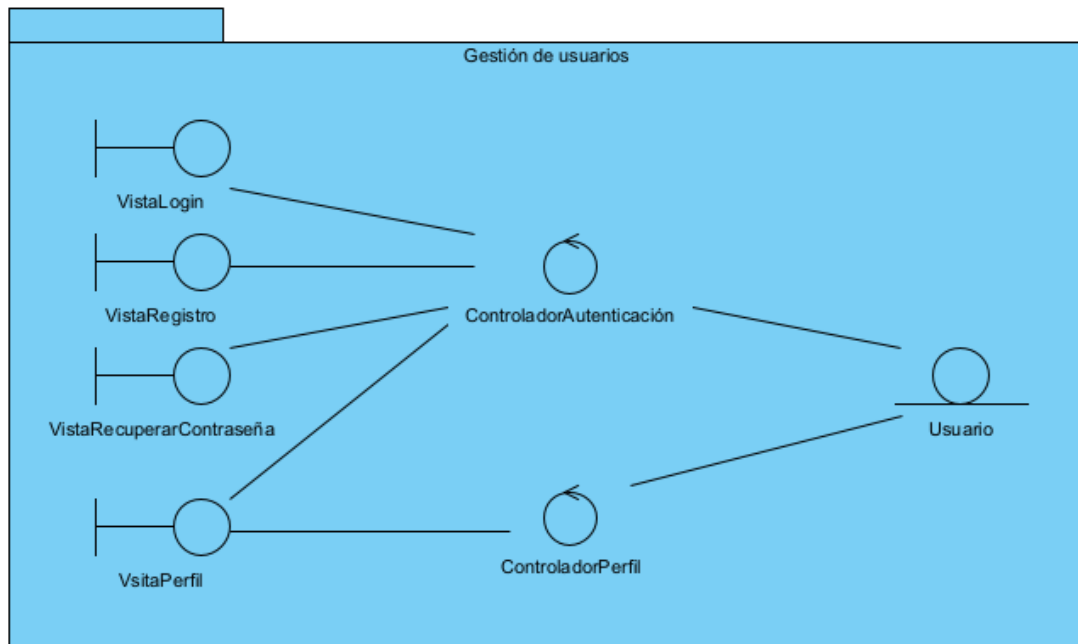


Figura 8: Diagrama de comunicación - Gestión de usuarios

Aplicación web de fórmula 1

5.5.4. Paquetes de análisis

Los paquetes de análisis albergarán las clases de análisis relacionadas se muestran y conforman el sistema.

En la *Figura 9* se muestra el diagrama de paquetes, con todos los paquetes que conforman el sistema:

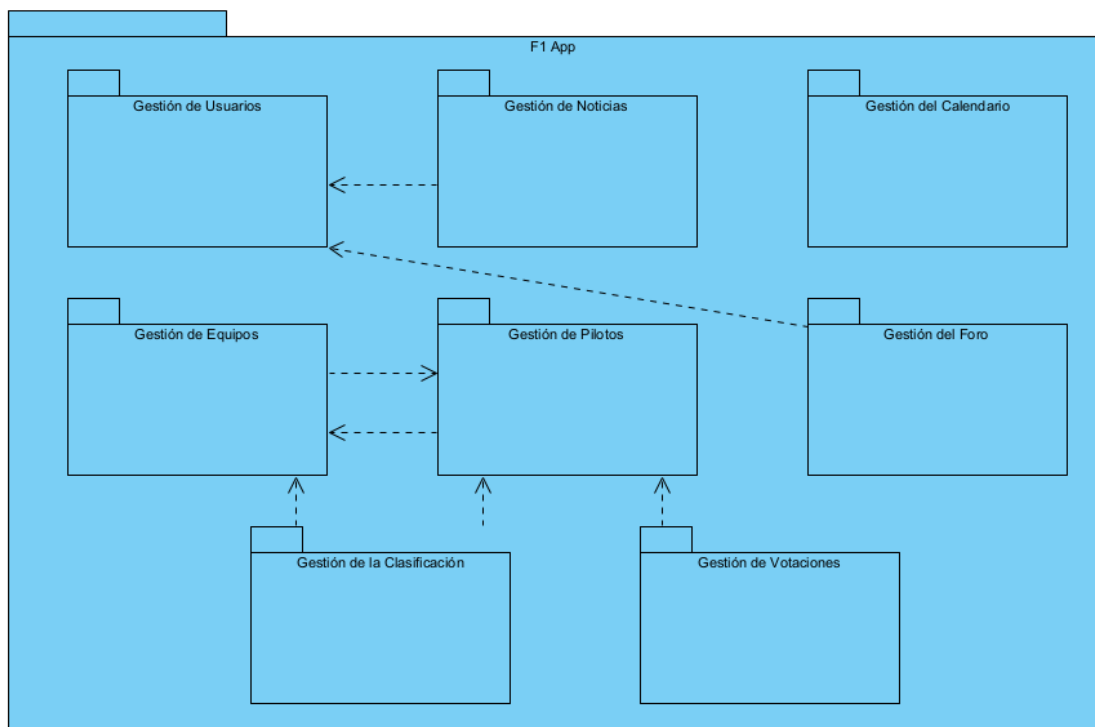


Figura 9: Diagrama de paquetes

5.5.5. Vista de arquitectura del modelo de análisis

En la vista de la arquitectura del modelo de análisis se agrupan cada clase dentro de la capa correspondiente en el patrón arquitectónico Modelo Vista Controlador (MVC), se puede ver la figura correspondiente en el **Anexo III – Análisis de requisitos**.

5.6. Diseño del sistema

En este apartado se recoge la fase de diseño del sistema, donde se busca una solución al problema presentado en la fase de búsqueda de requisitos. De modo que el resultado de esta fase será la base para comenzar con la implementación de la aplicación.

La documentación completa de este apartado se encuentra en el documento: **Anexo IV – Diseño del sistema software.**

5.6.1. Patrones arquitectónicos

5.6.1.1. Modelo-Vista-ModeloVista(MVVM)

El patrón Modelo-Vista-ModeloVista es un patrón de arquitectura de software, que se emplea para desacoplar lo máximo posible la interfaz de usuario de la lógica de la aplicación. Es similar al patrón Modelo-Vista-Controlador, pero en este caso el controlador es sustituido por el ModeloVista, el cual está enlazado con la vista mediante el Data Binding y realiza modificaciones sobre el modelo. En la *Figura 10* se puede ver un esquema de este patrón:

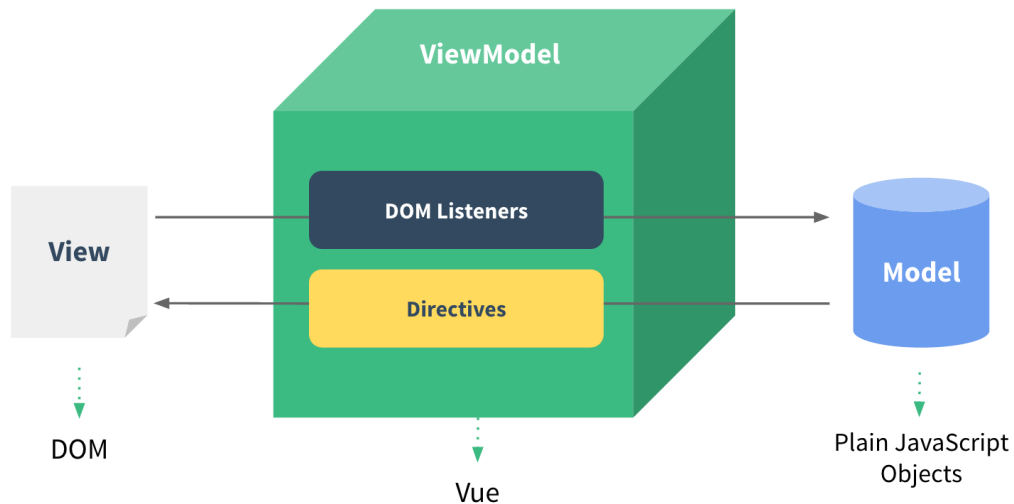


Figura 10: Patrón MVVM

El patrón se compone de tres capas:

- **Vista:** esta capa se encarga de representar la información a través de interfaces de usuario, las actualizaciones de visualización se activan automáticamente cuando cambien los datos.
- **ModeloVista:** esta capa se sitúa entre el modelo y la vista, se encarga de

Aplicación web de fórmula 1

sincronizar y enlazar la información del modelo a la vista. La comunicación entre la vista y el modelo de vista se realiza por medio los enlaces de datos.

- **Modelo:** es un objeto JavaScript simple ligeramente modificado que representa el modelo de los datos, conteniendo la información de los objetos. Siempre que los datos cambian, la vista se actualiza en el siguiente fotograma.

El enlace de datos bidireccional (Two-Ways Data Binding) es un protocolo mediante el cual cualquier cambio que se produzca en un objeto, se propaga a todas las capas en las que se encuentra, consiguiendo consistencia en todos los componentes. Es decir, mantiene los datos, variables, sincronizadas con la vista, de modo que si se modifica el valor de una variable se modificará en la vista y viceversa.

5.6.1.2. Publicador-Suscriptor

El patrón Publicador-Suscriptor (Publisher-Subscriber) es un patrón de intercambio de mensajes. Para ello se emplea un sistema de mensajería asíncrona que contenga:

- Un canal de mensajería de entrada utilizado por el publicador donde enviará los mensajes.
- Un canal de mensajería de salida utilizado por los suscriptores donde recibirán los mensajes.
- Un mecanismo para copiar cada mensaje del canal de entrada a los canales de salida.

En la *Figura 11* se puede ver un esquema de este patrón:

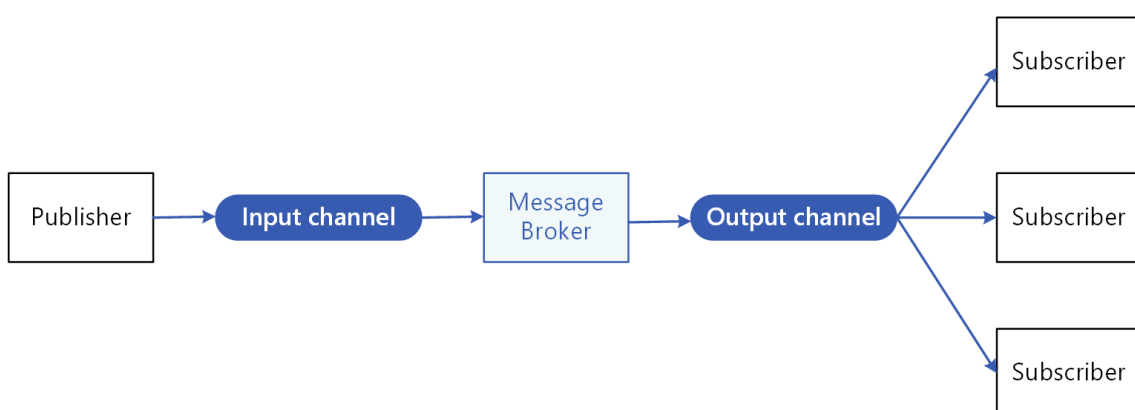


Figura 11: Patrón Publicador-Suscriptor

El servicio Cloud Firestore de Firebase es una base de datos NoSQL flexible, escalable y en la nube, cuyo fin es almacenar y sincronizar datos. Este servicio permite el acceso a los datos mediante suscripciones a cambios que se produzcan en las colecciones o los documentos almacenados.

Aplicación web de fórmula 1

5.6.2. Subsistemas de diseño

En este apartado se recogen los diagramas que representan las distintas partes del sistema, así como la relación entre los mismos. En la *Figura 12* se puede ver el diagrama de subsistemas de diseño.

El sistema se divide en 3 subsistemas relacionados:

- **Components:** contiene los ficheros Single File Components (SFC) de Vue.js, que siguen el patrón MVVM.
- **API:** abstrae a la base de datos del resto de funcionalidades de la aplicación.
- **Router:** establece una estructura de rutas que permite asignar los componentes de la aplicación a diferentes rutas de URL del navegador.

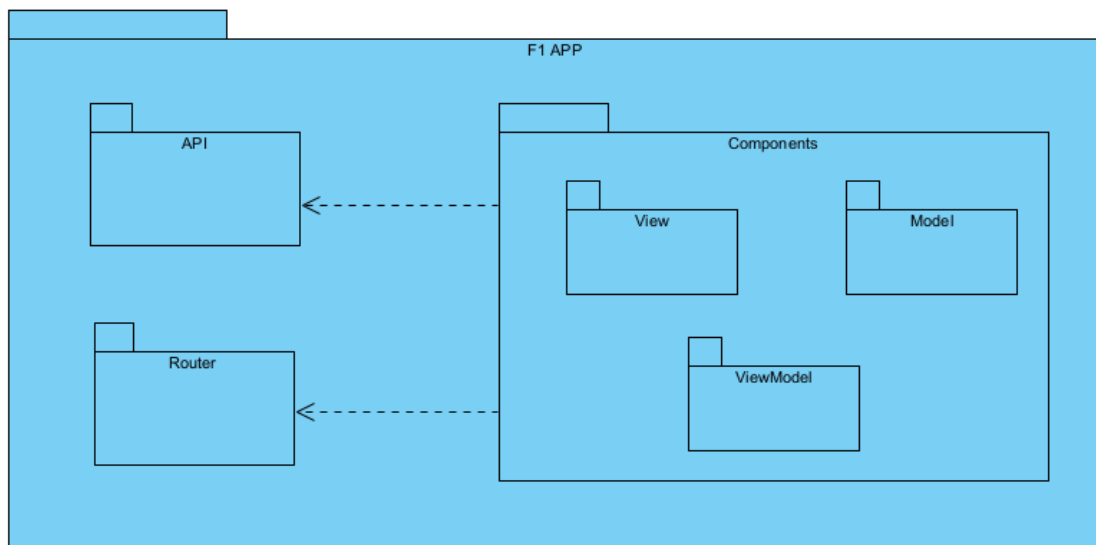


Figura 12: Diagrama de subsistemas de diseño

5.6.3. Clases de diseño

Los subsistemas de diseño están compuestos por otros paquetes o clases que se representan con diagramas. A continuación, en la *Figura 13* y en la *Figura 14* se muestra el diagrama del subsistema Components y del paquete Login:

Aplicación web de fórmula 1

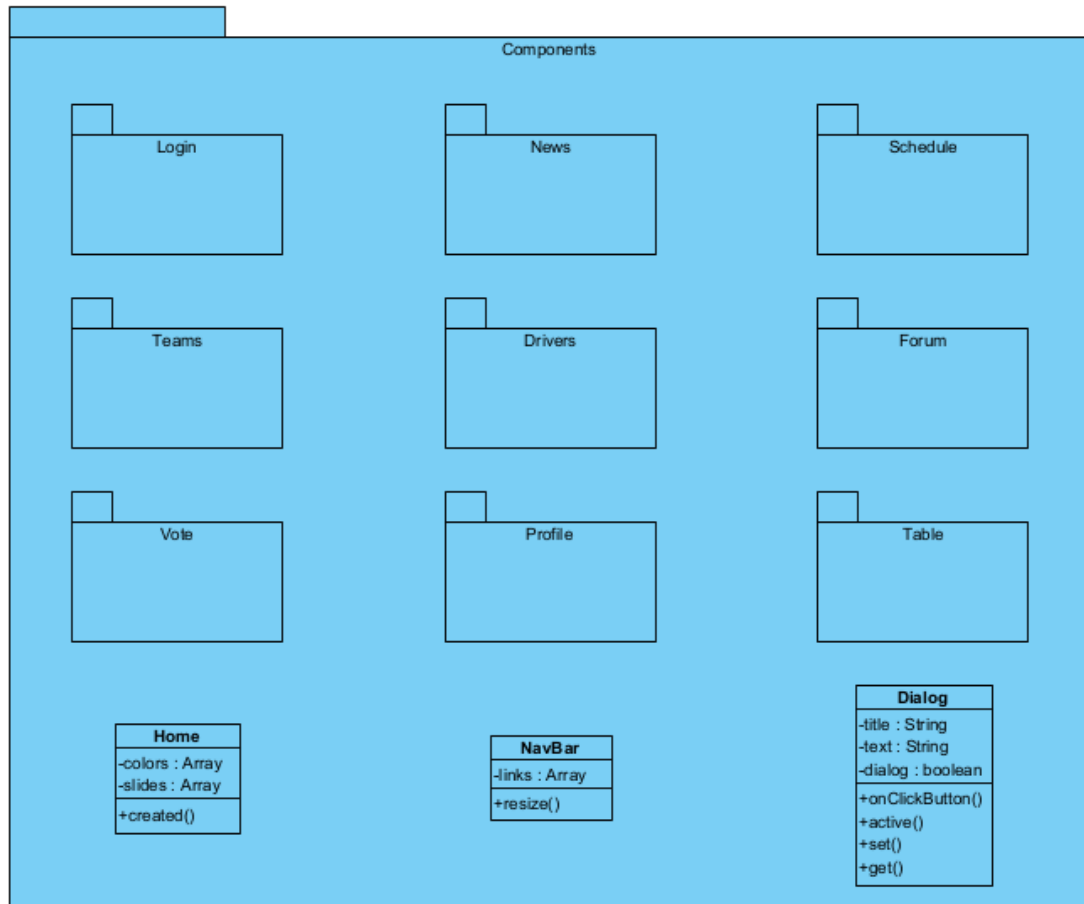


Figura 13: Diagrama del subsistema Components

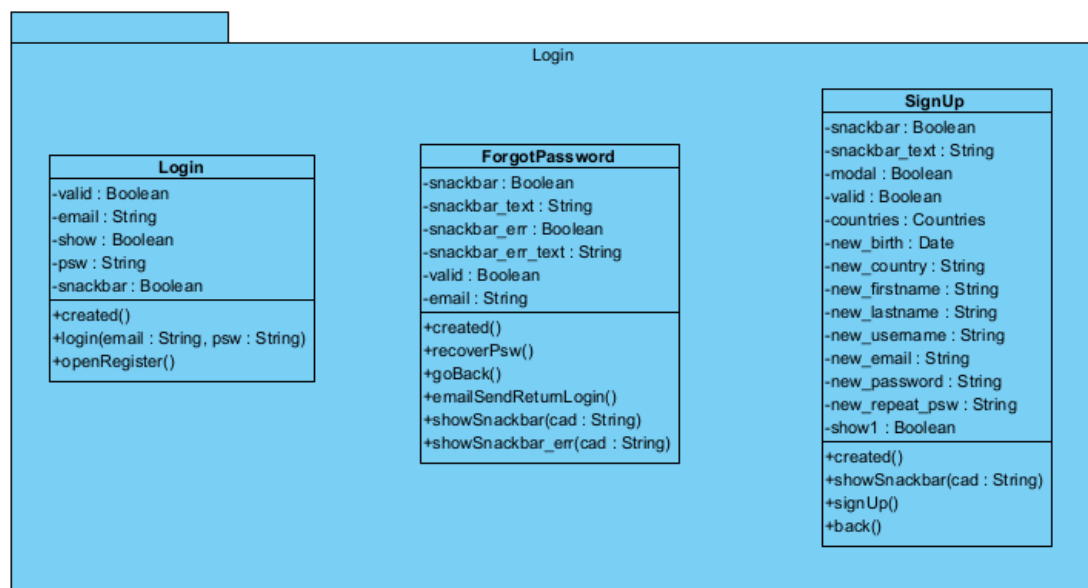


Figura 14: Diagrama del paquete Login

Aplicación web de fórmula 1

5.6.4. Vista arquitectónica del modelo de diseño

A continuación, en la *Figura 15* se muestra un diagrama de clases con los elementos que forman los Single File Components (SFC) de Vue.js siguiendo la organización del patrón MVVM.

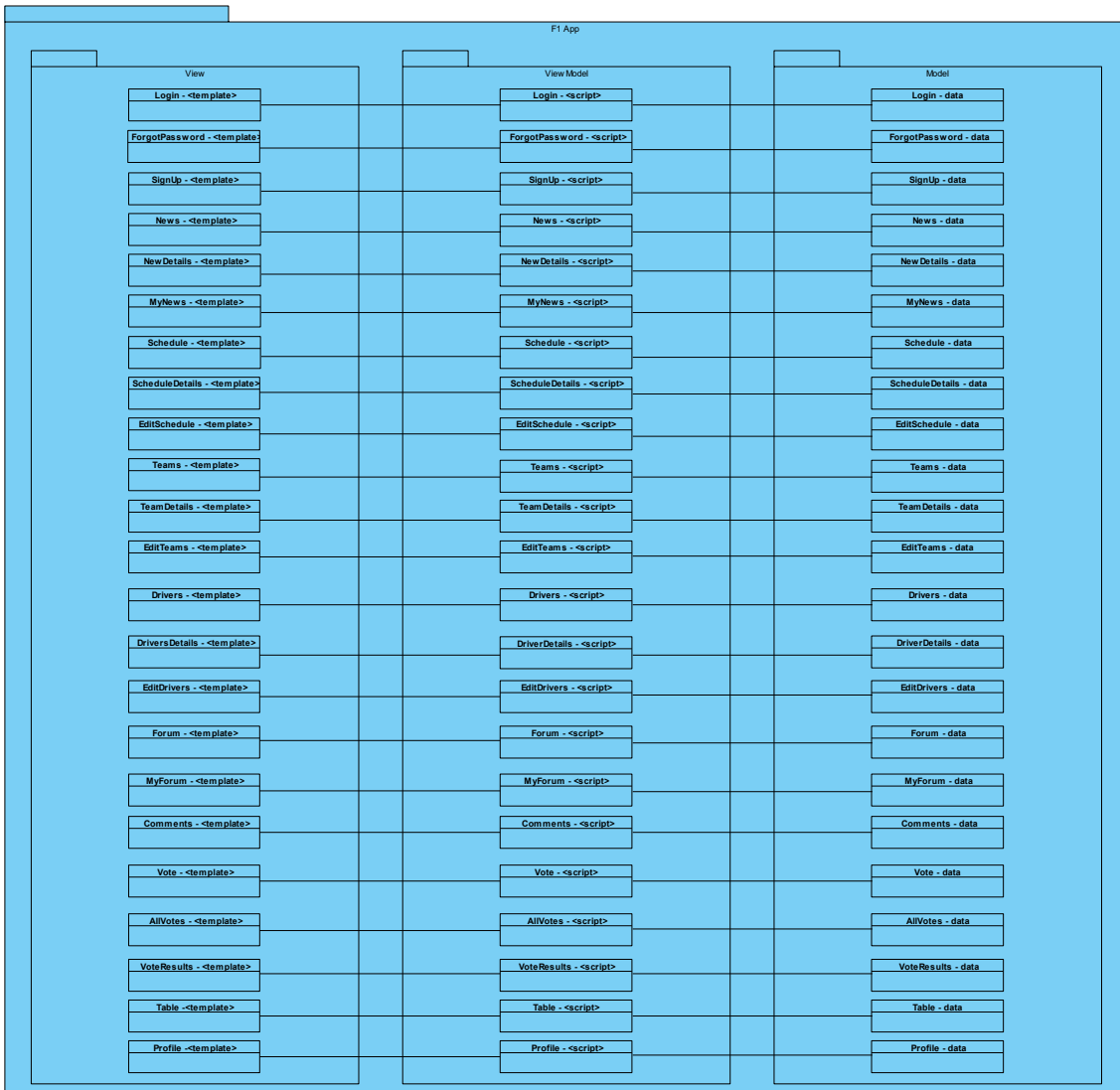


Figura 15: Vista arquitectónica del diseño

Aplicación web de fórmula 1

5.6.5. Realización de casos de uso

Para la realización de casos de uso se emplean diagramas de secuencia que indican los mensajes que se intercambian entre los diferentes componentes de la aplicación.

En la *Figura 16* se muestra el diagrama de secuencia del caso de uso Log in:

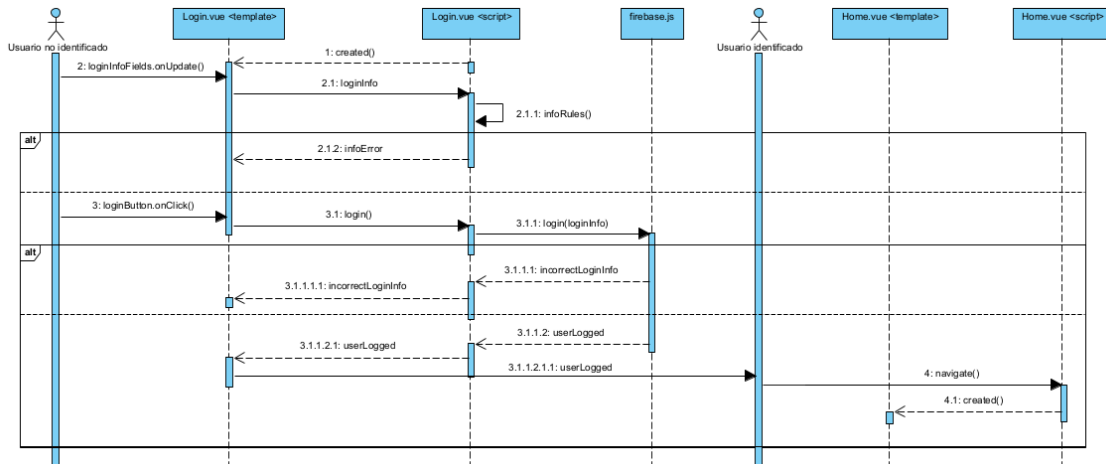


Figura 16: Diagrama de secuencia de diseño - Log in

5.6.6. Diseño de la base de datos

En la *Figura 17* muestra el diagrama del diseño de la base de datos NoSQL, donde se muestra la organización de la información que se va a almacenar en la aplicación:

Aplicación web de fórmula 1

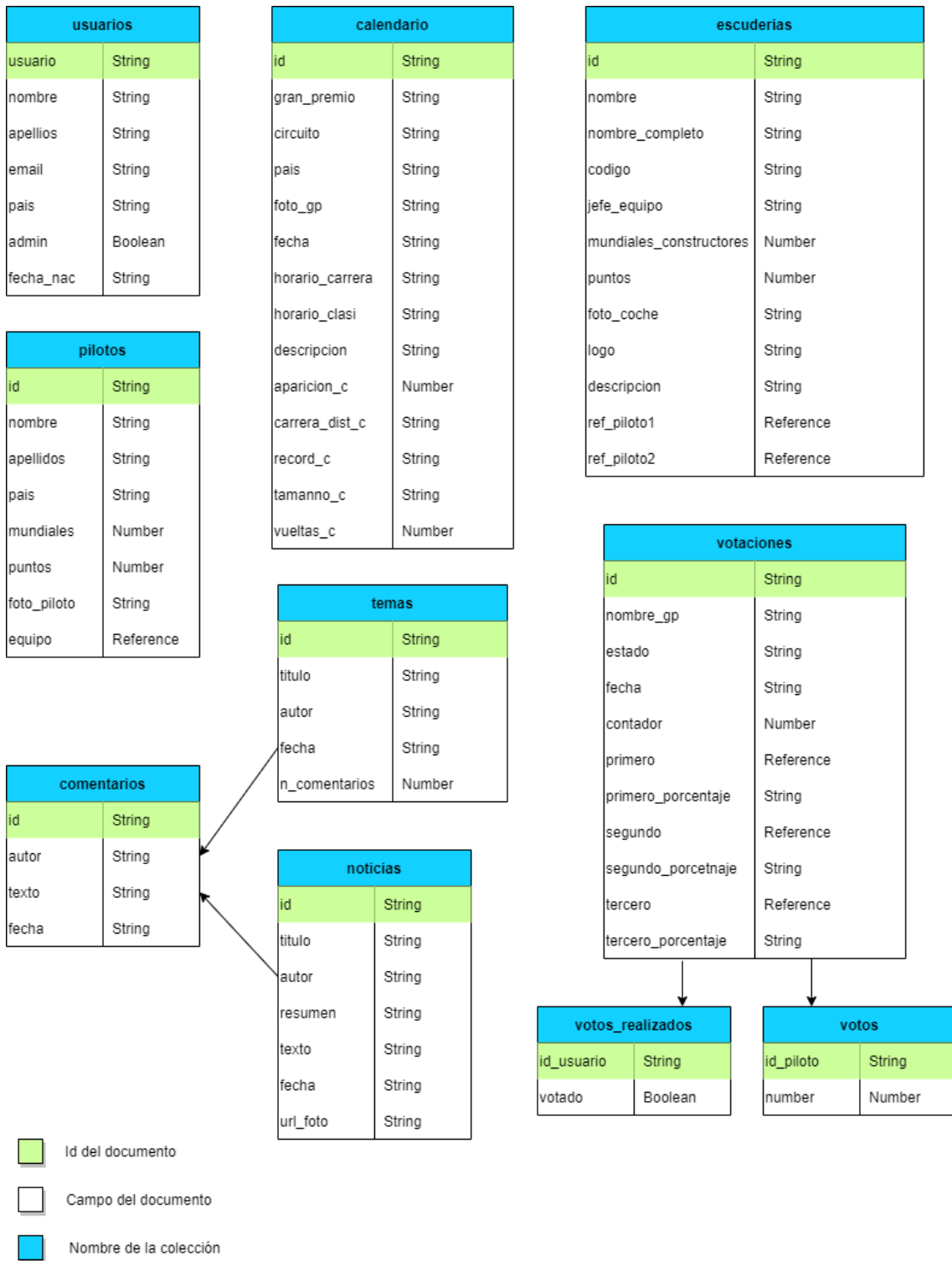


Figura 17: Diagrama de la base de datos

Cloud Firestore almacena los datos en documentos contenidos en colecciones. Además, estos pueden guardar subcolecciones con otros documentos, por ejemplo, los comentarios son subcolecciones que se crean dentro de cada documento de la

Aplicación web de fórmula 1

colección Noticias o Temas, al igual que votos y votos_realizados que son subcolecciones de votaciones.

5.6.7. Modelo de despliegue

En la *Figura 18* se muestra el modelo de despliegue, donde se exponen los nodos y componentes que forman el sistema. La aplicación está desplegada en el servicio de Hosting del servidor de Firebase, los usuarios accederán a través de cualquier navegador web y desde cualquier dispositivo:

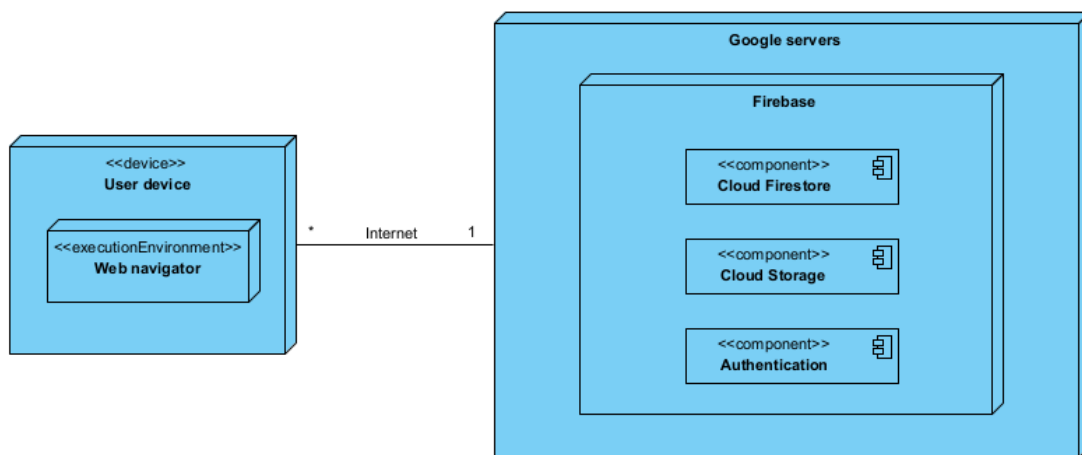


Figura 18: Diagrama de despliegue

6. Implementación

Para la implementación de la aplicación se ha empleado como guía los resultados de la fase de diseño y las herramientas comentadas previamente.

La documentación completa de este apartado se encuentra en el documento: **Anexo V – Documentación técnica**.

Para el desarrollo del sistema se comenzó por la configuración de la base de datos para poder registrar usuarios a través de la aplicación. Luego poco a poco se fueron añadiendo más funcionalidades de la aplicación y a su vez las colecciones necesarias en la base de datos.

6.1. Configuración de la base de datos

Para comenzar con el desarrollo, se implementó en Cloud Firestore la estructura de la base de datos que se propuso en el diseño, definiendo a su vez las reglas de acceso a ella.

Las colecciones definitivas que se emplean se pueden visualizar en la *Figura 19*:

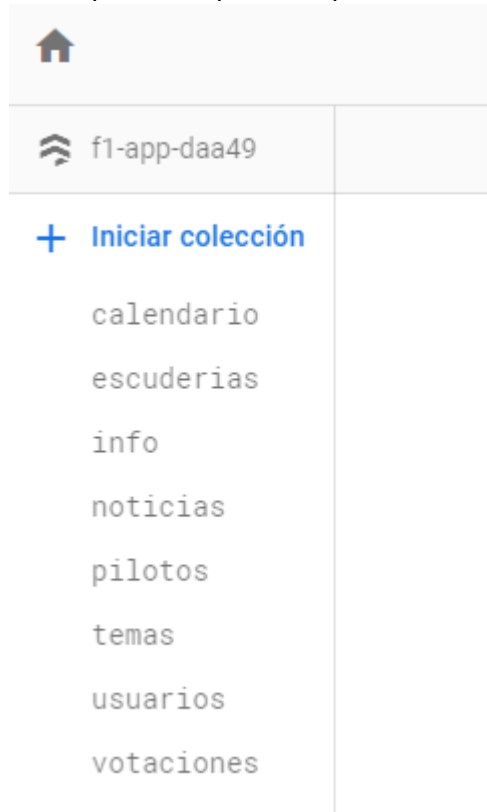


Figura 19: Colecciones de Cloud Firestore

Aplicación web de fórmula 1

- **calendario:** Contiene una serie de documentos donde cada uno almacena datos de un Gran Premio y puede cualquier usuario que haya iniciado sesión puede acceder a la colección.
- **escuderías:** Contiene documentos que representan a cada escudería y puede acceder cualquier usuario que haya iniciado sesión.
- **info:** Almacena el número del último Gran Premio del que se actualizo la clasificación y se emplea para ver si los datos obtenidos de la API externa no están almacenados en Cloud Firestore, si la API retorna un número de Gran Premio superior se guardan los datos recibidos de la API, pero si el número es igual o menor que el almacenado en esta colección los datos no se guardan porque ya están almacenados los últimos datos.
- **noticias:** Contiene documentos que representan a cada noticia publicada en el sistema y cada documento puede tener una subcolección donde se almacenan más documentos representando a cada comentario.
- **pilotos:** Contiene documentos que representan a cada piloto y puede acceder cualquier usuario que haya iniciado sesión.
- **temas:** Contiene documentos que representan a cada tema publicado en el sistema y cada documento tiene una subcolección donde se almacenan más documentos almacenando información de cada comentario.
- **usuarios:** Contiene documentos que representan a cada usuario. El ID de cada documento corresponde con el nombre de usuario.
- **votaciones:** Contiene documentos que representan a cada votación creada en el sistema, cada documento tiene 2 subcolecciones. Una almacena documentos de los usuarios que han votado y la otra almacena documentos con todos los pilotos del sistema y la cantidad de votos que han recibido.

6.2. Desarrollo de la aplicación web

Para el desarrollo de la aplicación web se han empleado las tecnologías citadas previamente en este documento:

- JavaScript: empleado como lenguaje de programación.
- Vue.js: framework utilizado para el desarrollo de la aplicación web
- Vuetify.js: framework empleado para el desarrollo de interfaces de usuario siguiendo las convenciones del Material Design.
- Firebase: servicio de Google que se ha utilizado como base de datos NoSQL, sistemas de autenticación de usuarios, servicio de almacenamiento de contenido multimedia y sitio de alojamiento de la aplicación.

Como se ha comentado anteriormente para el desarrollo de aplicación se fue poco a poco añadiendo cada una de las funcionalidades de la aplicación, comenzando por el acceso de los usuarios al sistema, continuando por la gestión de las noticias, calendario, escuderías... Por último, se implemento la actualización automática de la clasificación mediante peticiones a una API externa que proporciona los puntos de

Aplicación web de fórmula 1

cada escudería y pilotos de la competición.

6.3. Despliegue de la aplicación

Durante la fase de desarrollo del proyecto el despliegue de la aplicación se realizó de manera local, debido a la rapidez de actualización de cualquier cambio que se realizaba en el código sobre la aplicación desplegada localmente.

Una vez realizada la correcta implementación de todas las funcionalidades de la aplicación se empleó el servicio Hosting de Firebase para realizar el despliegue. Para acceder a la última versión desplegada de la aplicación:

- <https://f1-app-daa49.firebaseio.com/>
- <https://f1-app-daa49.web.app/>

6.4. Documentación técnica

Para consultar cualquier aspecto del código fuente o su estructura en más detalle se puede encontrar la información en el documento **Anexo V: Documentación técnica**.

Cabe destacar que también se han generado unos archivos con la documentación de cada componente de Vue.js y funciones de JavaScript empleadas.

7. Pruebas

Durante el desarrollo de cada subsistema de la aplicación se realizaron pruebas sobre los mismos. A medida que se iba añadiendo una nueva funcionalidad en la aplicación, se realizaban pruebas para comprobar su funcionamiento y una vez se tenía el subsistema completo se realizaban pruebas unitarias sobre cada uno de los componentes que formaban el subsistema.

Según se iban añadiendo más subsistemas aparte de las pruebas unitarias a sus componentes, también se realizaban pruebas de integración del nuevo subsistema con los implementados previamente que ya habían sido probados, pero las pruebas en este caso permitían comprobar que las nuevas actualizaciones no afectaban al correcto funcionamiento previo de la aplicación.

Con todos los subsistemas implementados se realizaron pruebas globales, comprobando la navegación entre diferentes vistas, la responsividad de cada una de las vistas en diferentes dispositivos y la funcionalidad global de la aplicación.

Por último, se realizaron algunas pruebas de seguridad, rendimiento, etc. Con el objetivo de comprobar que no había alguna brecha importante de seguridad y que los tiempos de respuesta de la aplicación eran aceptables.

8. Funcionalidad de la aplicación

La funcionalidad de la aplicación se divide en los siguientes apartados:

- Acceso a la aplicación
- Noticias
- Calendario
- Escuderías
- Pilotos
- Foro
- Votaciones
- Clasificación
- Perfil

La documentación completa de este apartado se encuentra en el documento: **Anexo VI – Manual de usuario.**

8.1. Acceso a la aplicación

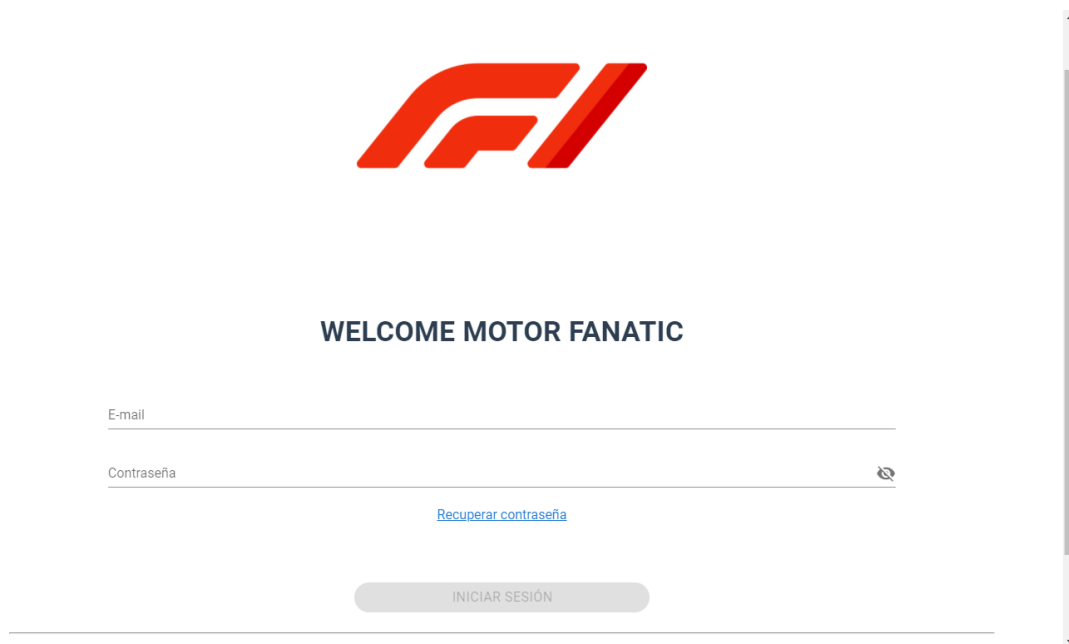
8.1.1. Inicio

Para poder acceder a la aplicación web el usuario tiene que dirigirse a las siguientes direcciones URLs:

- <https://f1-app-daa49.firebaseio.com/>
- <https://f1-app-daa49.web.app/>

Si no se tiene una sesión iniciada, la aplicación redireccionará a la página de inicio de sesión mostrada en la *Figura 20*.

Aplicación web de fórmula 1

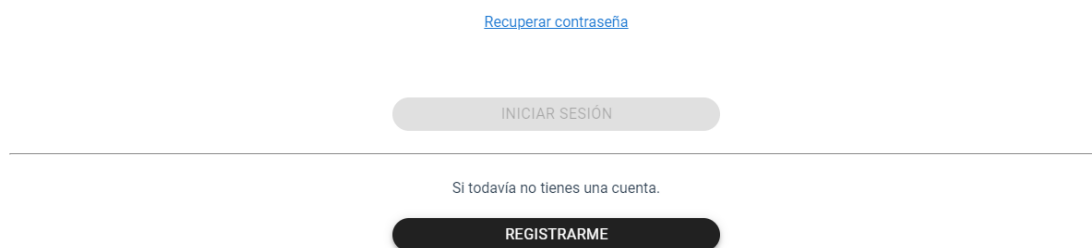


The screenshot shows the login interface for 'WELCOME MOTOR FANATIC'. At the top center is a red stylized 'F1' logo. Below it, the text 'WELCOME MOTOR FANATIC' is displayed in a bold, dark blue font. There are two input fields: 'E-mail' and 'Contraseña'. The 'Contraseña' field has a small eye icon to its right. Below the input fields is a blue link that says 'Recuperar contraseña'. At the bottom of the form is a grey button labeled 'INICIAR SESIÓN'.

Figura 20: Inicio de sesión

8.1.2. Registrarse

Si el usuario todavía no tiene una cuenta creada, debe dirigirse a la parte inferior de la página de inicio de sesión y seleccionar “Registrarme” como se muestra en la Figura 21.



This screenshot shows the bottom portion of the login page. It features a blue link 'Recuperar contraseña' at the top. Below it is a grey button labeled 'INICIAR SESIÓN'. At the bottom, there is a line of text that reads 'Si todavía no tienes una cuenta.' followed by a black button labeled 'REGISTRARME'.

Figura 21: Inicio de sesión – Registrarse

La aplicación redireccionará al usuario a la página de registro donde tendrá que introducir los datos necesarios para registrarse (Figura 22). El botón de “Aceptar” no se activará hasta que todos los datos sean válidos, durante el proceso de registro la aplicación informará al usuario en todo momento de cualquier error. En caso de que el usuario desee volver a la página de inicio de sesión debe seleccionar “Cancelar”.

Aplicación web de fórmula 1

REGISTRARSE
Por favor, rellene todos los campos para crear una cuenta.

Nombre

Apellidos

Pais

Fecha de nacimiento

Nombre de usuario

E-mail

Contraseña

Replta la contraseña

Figura 22: Registrarse

8.1.3. Recuperar contraseña

Si un usuario registrado en el sistema no recuerda su contraseña para acceder, debe dirigirse a la parte inferior de la página de inicio de sesión y seleccionar “Recuperar contraseña” para obtener una nueva.

La aplicación redireccionará al usuario a la página de recuperar contraseña donde tendrá que introducir su email como se muestra en la *Figura 23*. El botón de “Enviar” no se activará hasta que email sea válido. En caso de que el usuario desee volver a la página de inicio de sesión debe seleccionar “Cancelar”.

Aplicación web de fórmula 1



The screenshot shows a web interface for password recovery. At the top center is a red stylized 'F1' logo. Below it, the heading 'RECUPERAR CONTRASEÑA' is displayed in bold blue text. Underneath the heading is a small instruction: 'Por favor, introduzca su email para recuperar la contraseña.' Below this is a horizontal line representing an email input field, with the placeholder text 'E-mail' on the left. At the bottom of the form are two buttons: a black button labeled 'CANCELAR' and a grey button labeled 'ENVIAR'.

Figura 23: Recuperar contraseña

La aplicación enviará un correo al usuario con un enlace para cambiar la contraseña. Una vez que el usuario introduzca la nueva contraseña podrá acceder a la aplicación con dicha contraseña.

8.2. Noticias

8.2.1. Visualización de noticias

Una vez que el usuario inicie sesión en la aplicación tendrá acceso a las noticias de la aplicación donde podrá visualizarlas como se muestra en la *Figura 24*, para acceder a las noticias el usuario debe seleccionar “News” en la barra de navegación.

Aplicación web de fórmula 1

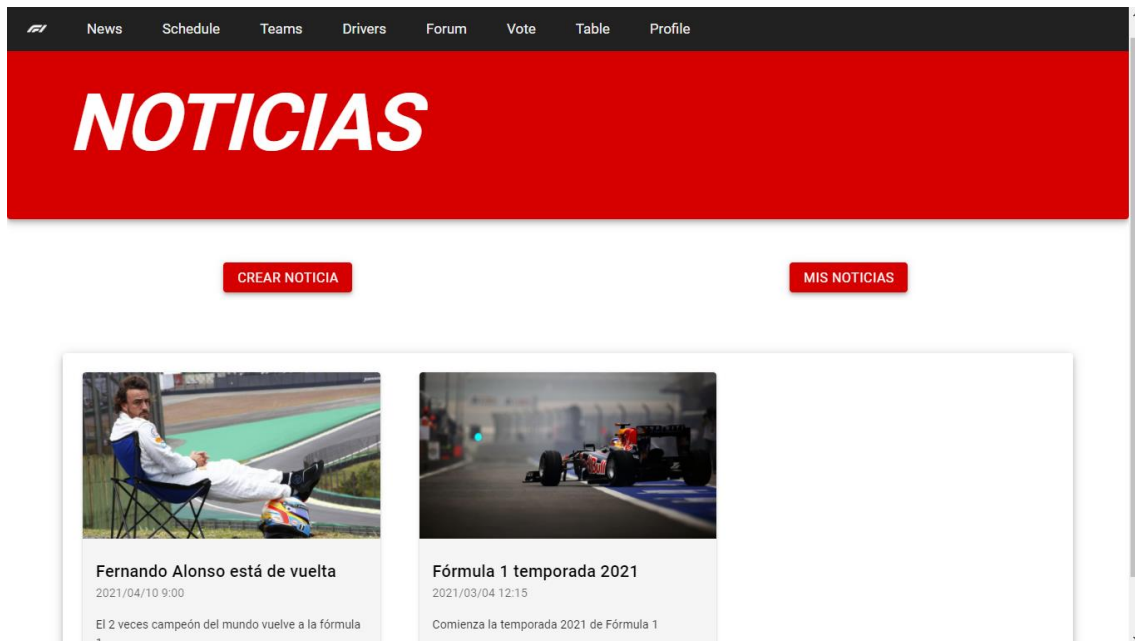


Figura 24: Noticias

El usuario podrá seleccionar una noticia para visualizar más contenido acerca de ella y los comentarios que hayan realizado otros usuarios sobre esa noticia.

8.2.2. Gestión de los comentarios de una noticia

Para realizar un comentario sobre una noticia el usuario debe seleccionar la opción "COMENTAR" que permitirá al usuario introducir sus comentarios como se muestra en la Figura 25.

Aplicación web de fórmula 1

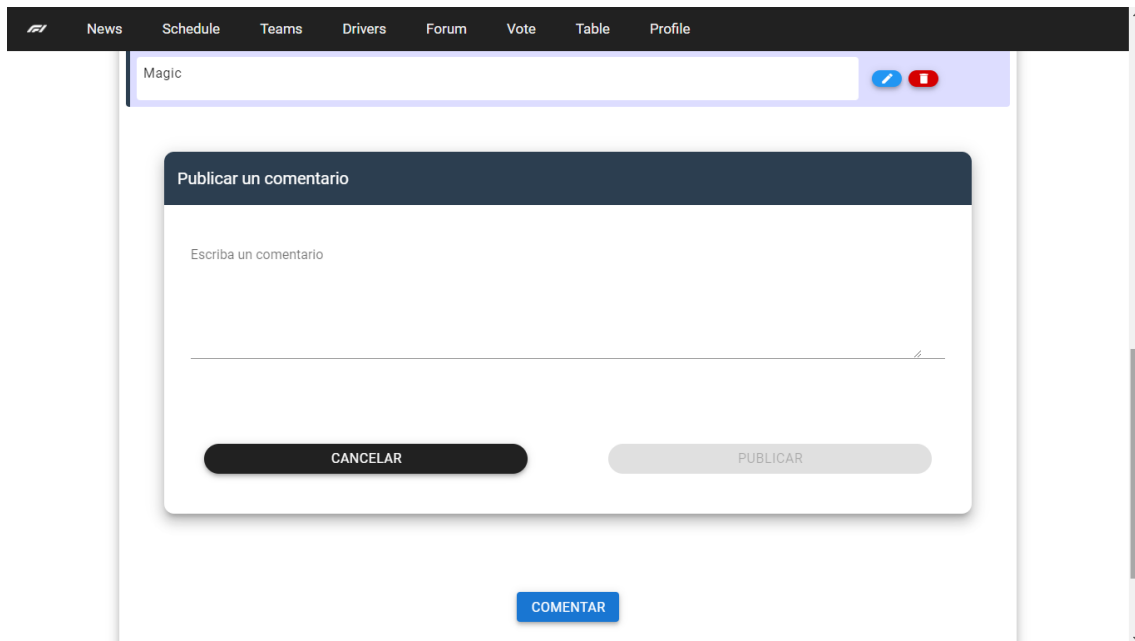




Figura 25: Publicar comentario en una noticia

Para que un usuario edite cualquier comentario que haya realizado debe seleccionar el botón . La aplicación mostrará el texto actual del comentario y el usuario podrá modificarlo y seleccionar “GUARDAR” para confirmar el cambio o seleccionar “CANCELAR” para descartar el cambio.

Para que un usuario elimine cualquier comentario que haya realizado debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.2.3. Gestión de las noticias

Para los usuarios que tengan permisos de administrador en la aplicación visualizarán los botones “CREAR NOTICIA” y “MIS NOTICIAS” como se muestran en la Figura 26, donde dispondrán de la opción de crear, editar o eliminar noticias en la aplicación.

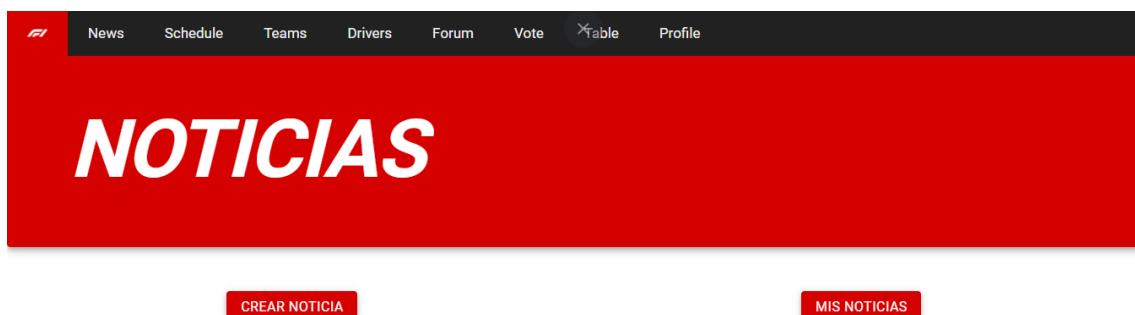
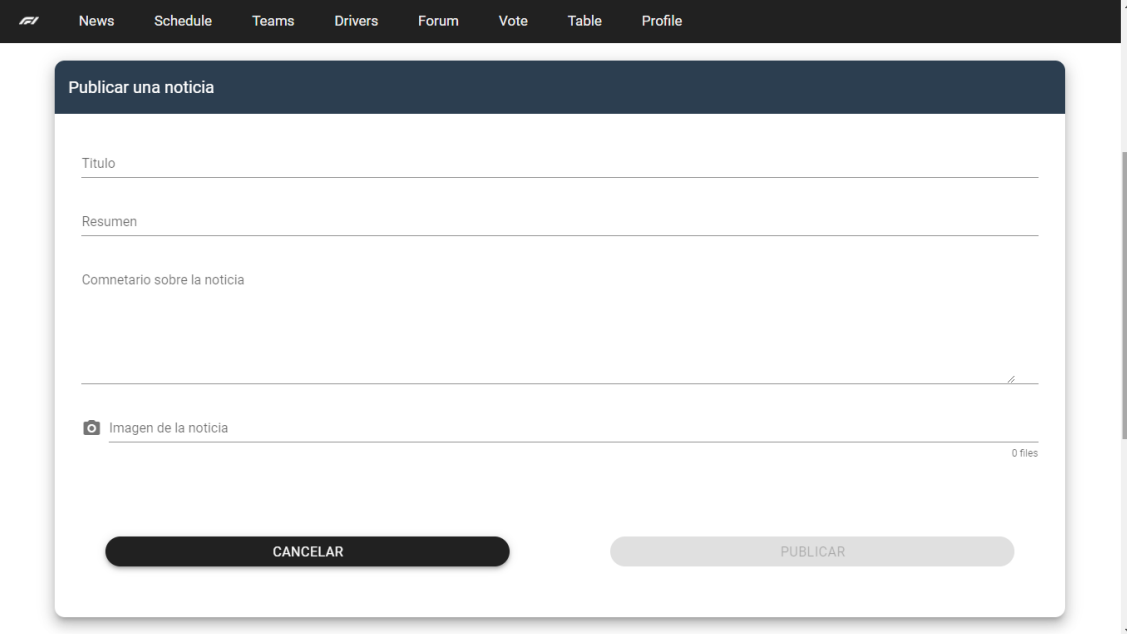


Figura 26: Botones para gestionar las noticias

Aplicación web de fórmula 1

Para que un usuario con permisos de administrador publique una noticia debe seleccionar la opción “CREAR NOTICIA”. La aplicación desplegará un formulario con los datos que se deben rellenar como se muestra en la *Figura 27*.



The screenshot shows a web application interface with a dark navigation bar at the top containing the following menu items: News, Schedule, Teams, Drivers, Forum, Vote, Table, and Profile. Below the navigation bar is a modal window titled "Publicar una noticia". The form contains the following fields and elements:

- Título**: A text input field.
- Resumen**: A text input field.
- Comentario sobre la noticia**: A text input field.
- Imagen de la noticia**: A file upload field with a camera icon, currently showing "0 files".
- Buttons**: Two buttons at the bottom: "CANCELAR" (dark grey) and "PUBLICAR" (light grey).


Figura 27: Publicar noticia


Para que un usuario con permisos de administrador visualice todas las noticias que ha publicado debe seleccionar la opción “MIS NOTICIAS”. La aplicación le redireccionará la página de mis noticias como se muestra en la *Figura 28*.

Aplicación web de fórmula 1



Figura 28: Mis noticias

Para que un usuario con permisos de administrador edite una noticia debe seleccionar el botón . La aplicación mostrará un formulario con la información actual de la noticia y el usuario podrá modificarla y seleccionar “ACTUALIZAR” para confirmar los cambios o seleccionar “CANCELAR” para descartar los cambios.

Para que un usuario elimine cualquier noticia que haya realizado debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.3. Calendario

8.3.1. Visualización del calendario

Una vez que el usuario inicie sesión en la aplicación tendrá acceso al calendario de la competición de Fórmula 1 como se muestra en la *Figura 29*, para acceder a el calendario el usuario debe seleccionar “Schedule” en la barra de navegación.

Aplicación web de fórmula 1

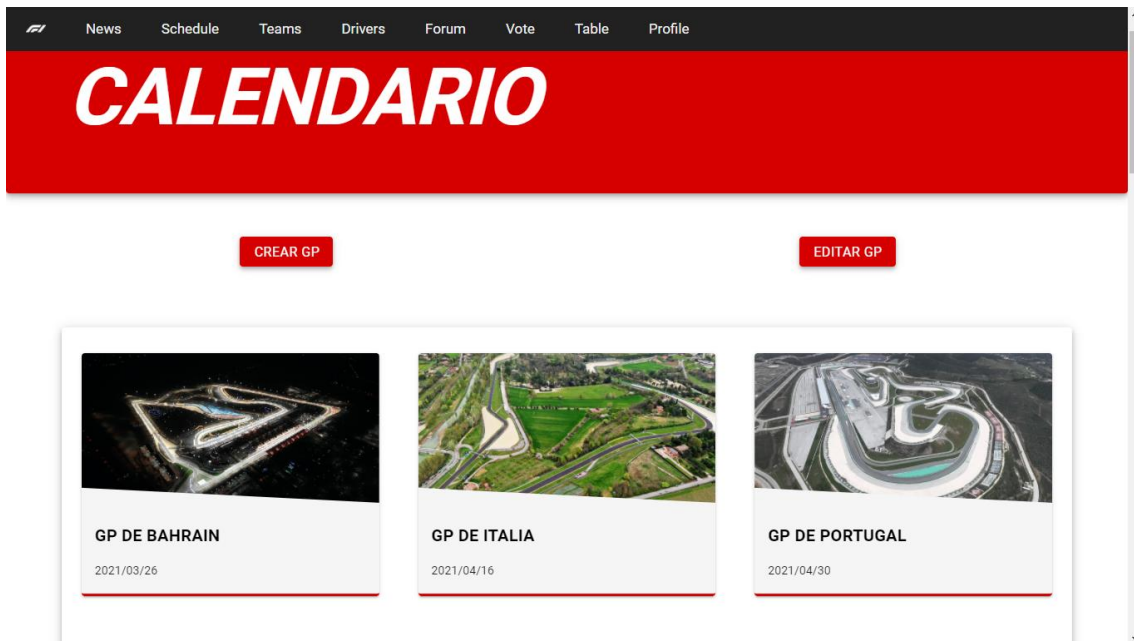


Figura 29: Calendario

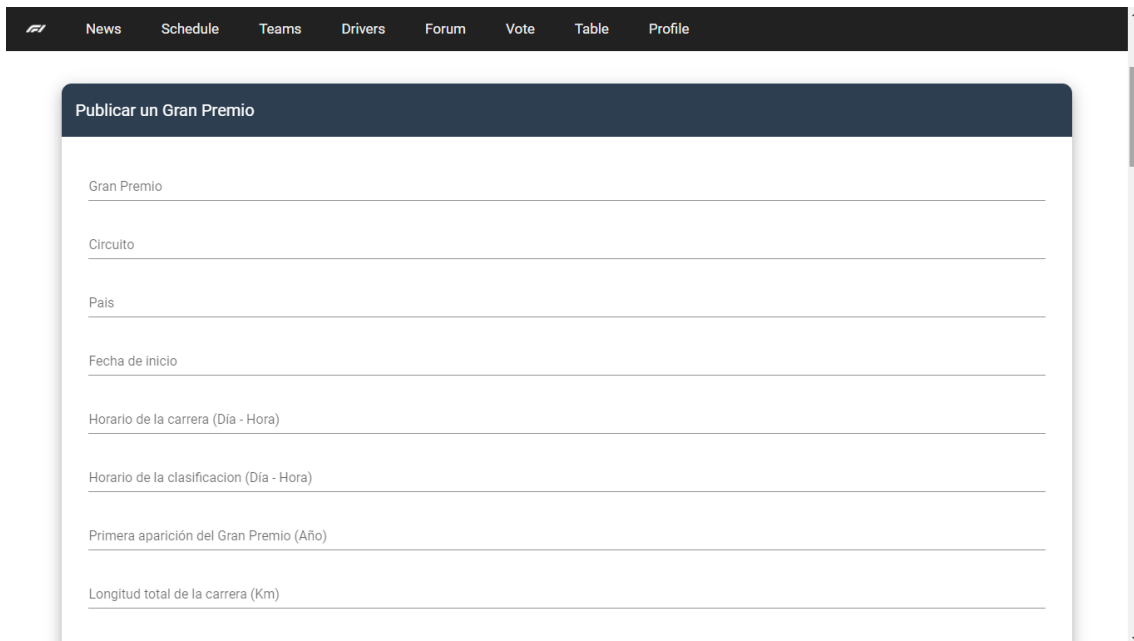
El usuario podrá seleccionar un Gran Premio para visualizar más contenido acerca de él.

8.3.2. Gestión del calendario

Para los usuarios que tengan permisos de administrador en la aplicación visualizarán los botones "CREAR GP" y "EDITAR GP", y dispondrán de la opción de crear, editar o eliminar Grandes Premios en la aplicación.

Para que un usuario con permisos de administrador publique un Gran Premio debe seleccionar la opción "CREAR GP". La aplicación desplegará un formulario con los datos que se deben rellenar como se muestra en la *Figura 30*.

Aplicación web de fórmula 1



The screenshot shows a web application interface with a dark navigation bar at the top containing the F1 logo and menu items: News, Schedule, Teams, Drivers, Forum, Vote, Table, and Profile. Below the navigation bar is a form titled 'Publicar un Gran Premio' with a dark header. The form contains several input fields for creating a Grand Prix entry:

- Gran Premio
- Circuito
- Pais
- Fecha de inicio
- Horario de la carrera (Día - Hora)
- Horario de la clasificacion (Día - Hora)
- Primera aparición del Gran Premio (Año)
- Longitud total de la carrera (Km)

Figura 30: Publicar Gran Premio

Para que un usuario con permisos de administrador edite o elimine algún Gran Premio que hay creado debe seleccionar la opción “EDITAR GP”. La aplicación lo redireccionará a la página de editar Grandes Premios como se muestra en la *Figura 31*.

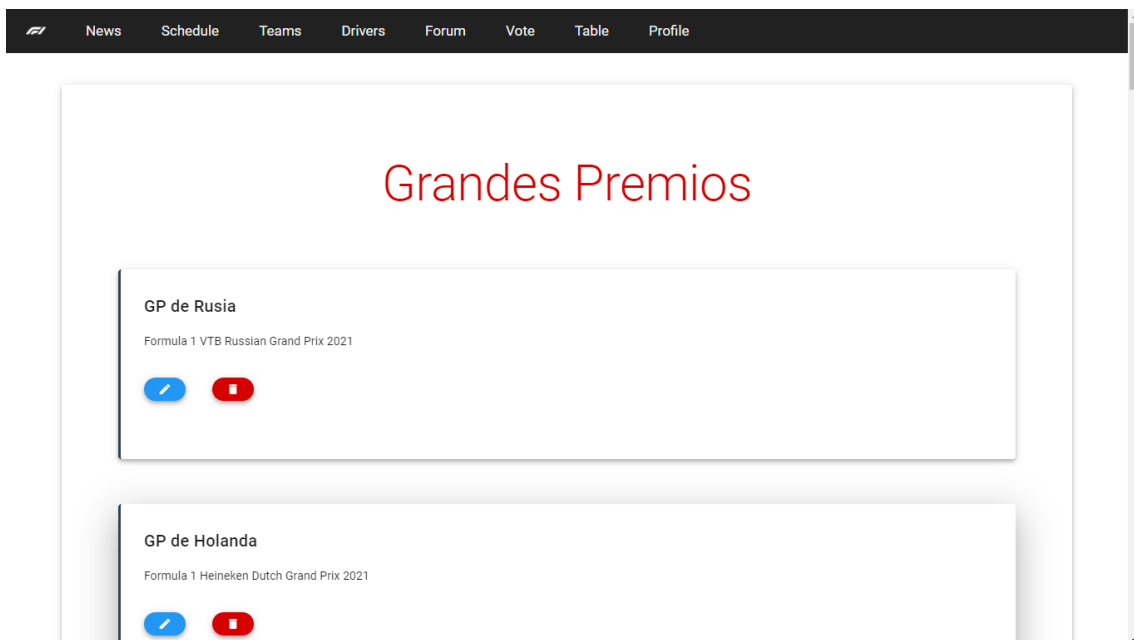




Figura 31: Grandes Premios

Aplicación web de fórmula 1

Para que un usuario con permisos de administrador edite un Gran Premio debe seleccionar el botón . La aplicación mostrará un formulario con la información actual del Gran Premio y el usuario podrá modificarla y seleccionar “ACTUALIZAR” para confirmar los cambios o seleccionar “CANCELAR” para descartar los cambios.

Para que un usuario elimine un Gran Premio debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.4. Escuderías

8.4.1. Visualización de las escuderías

Una vez que el usuario inicie sesión en la aplicación tendrá acceso las escuderías de la competición de Fórmula 1 como se muestra en la *Figura 32*, para acceder a las escuderías el usuario debe seleccionar “Teams” en la barra de navegación.

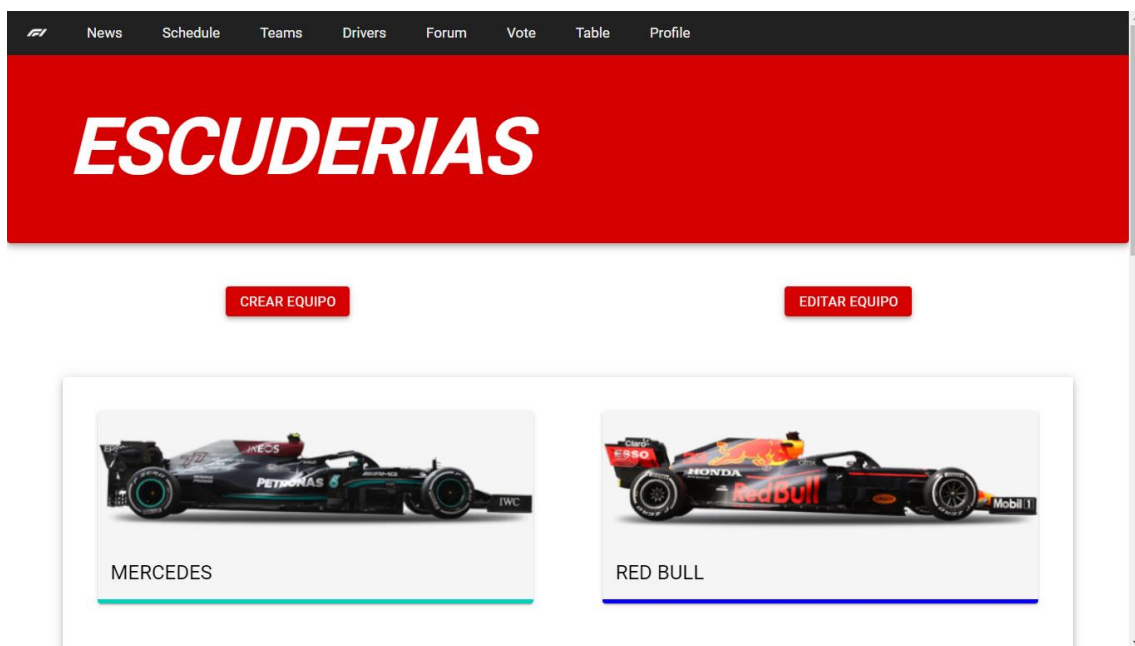


Figura 32: Escuderías

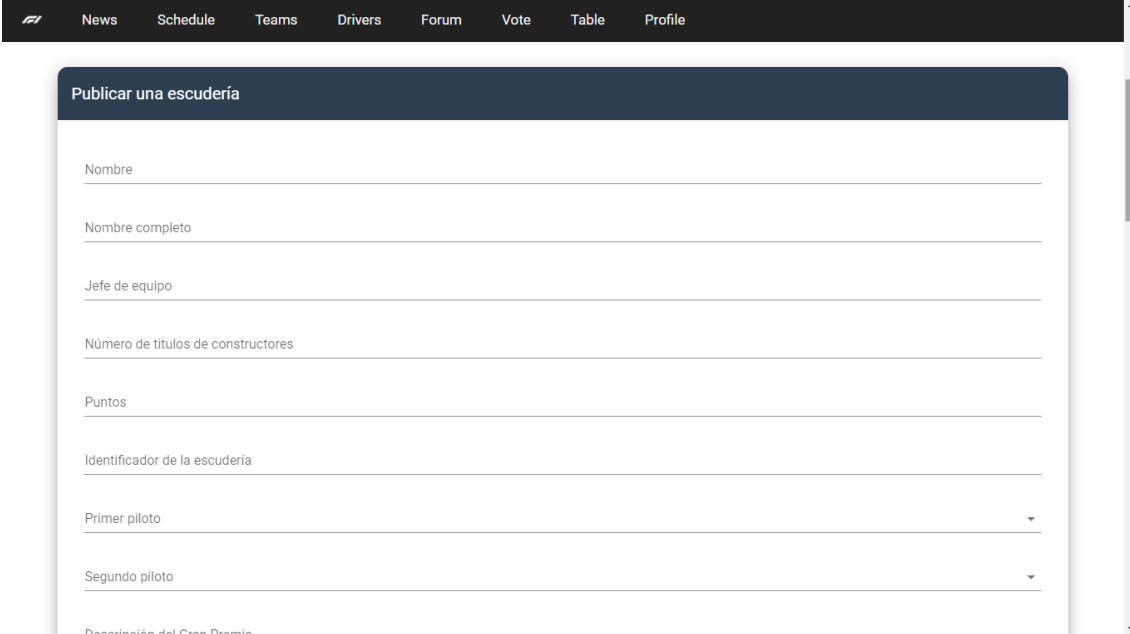
El usuario podrá seleccionar una escudería para visualizar más contenido acerca de ella.

8.4.2. Gestión de las escuderías

Para los usuarios que tengan permisos de administrador en la aplicación visualizarán los botones “CREAR EQUIPO” y “EDITAR EQUIPO”, y dispondrán de la opción de crear, editar o eliminar escuderías en la aplicación.

Aplicación web de fórmula 1

Para que un usuario con permisos de administrador publique una escudería debe seleccionar la opción “CREAR EQUIPO”. La aplicación desplegará un formulario con los datos que se deben rellenar como se muestra en la *Figura 33*.



The image shows a screenshot of a web application interface for publishing a team. At the top, there is a dark navigation bar with the F1 logo and menu items: News, Schedule, Teams, Drivers, Forum, Vote, Table, and Profile. Below this is a white form titled 'Publicar una escudería' with a dark blue header. The form contains several input fields: 'Nombre', 'Nombre completo', 'Jefe de equipo', 'Número de títulos de constructores', 'Puntos', 'Identificador de la escudería', 'Primer piloto' (with a dropdown arrow), and 'Segundo piloto' (with a dropdown arrow). At the bottom of the form, there is a small text label 'Descripción del Gran Premio'.

Figura 33: Publicar escuderías

Para que un usuario con permisos de administrador edite o elimine alguna escudería que hay creada debe seleccionar la opción “EDITAR EQUIPO”. La aplicación le redireccionará la página de editar escuderías como se muestra en la *Figura 34*.

Aplicación web de fórmula 1

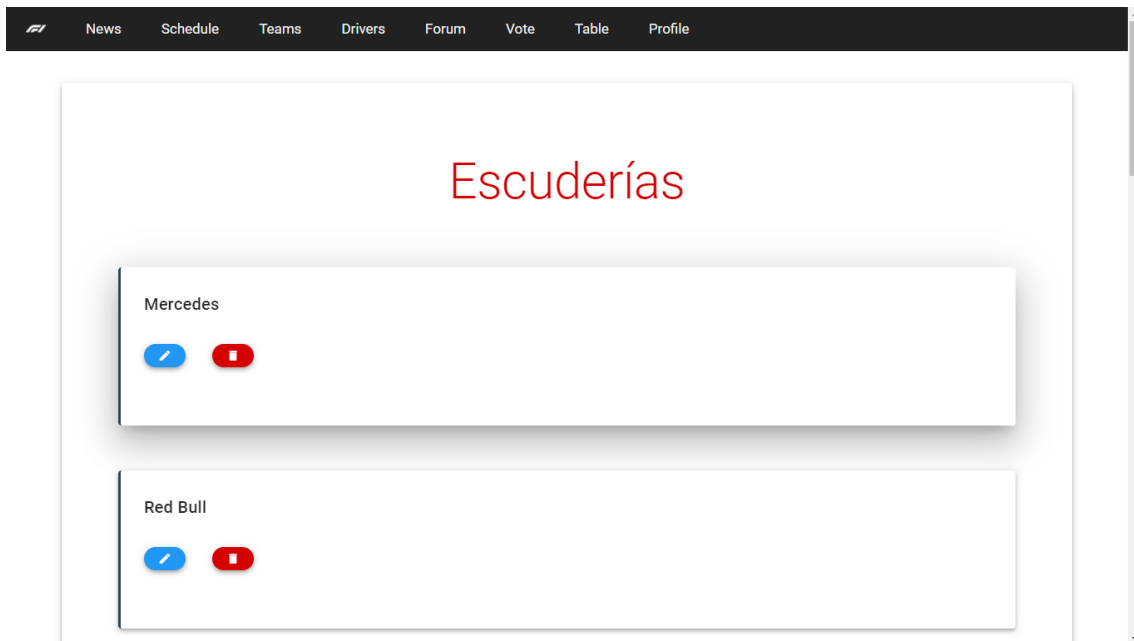




Figura 34: Escuderías

Para que un usuario con permisos de administrador edite una escudería debe seleccionar el botón . La aplicación mostrará un formulario con la información actual de la escudería y el usuario podrá modificarla y seleccionar “ACTUALIZAR” para confirmar los cambios o seleccionar “CANCELAR” para descartar los cambios.

Para que un usuario elimine una escudería debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.5. Pilotos

8.5.1. Visualización de los pilotos

Una vez que el usuario inicie sesión en la aplicación tendrá acceso los pilotos de la competición de Fórmula 1 como se muestra en la *Figura 35*, para acceder a los pilotos el usuario debe seleccionar “Drivers” en la barra de navegación.

Aplicación web de fórmula 1

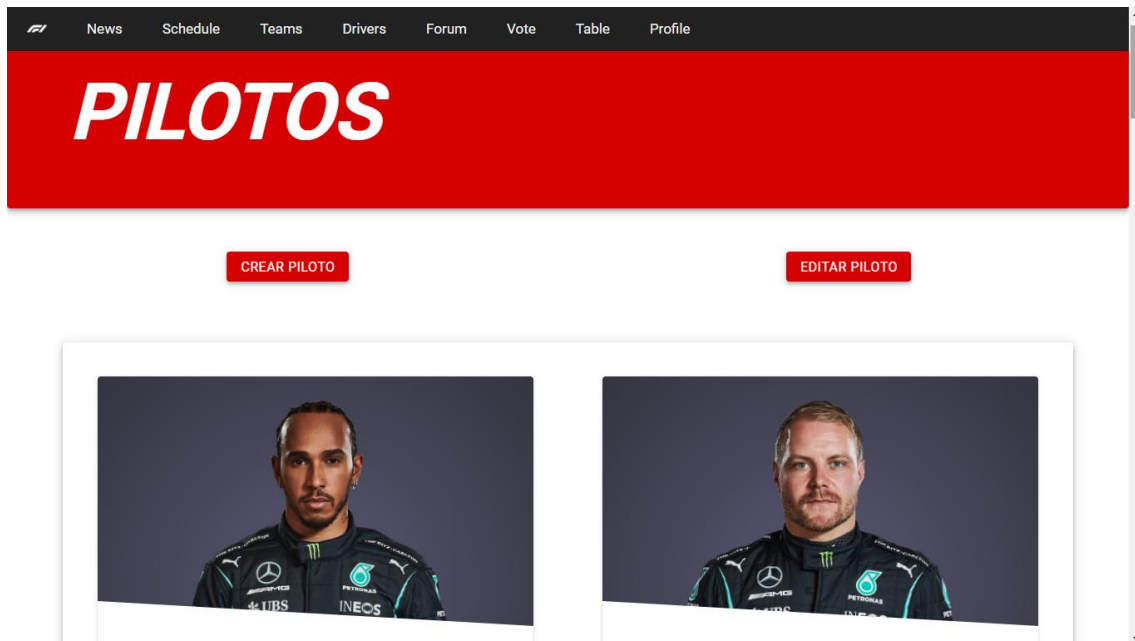


Figura 35: Pilotos

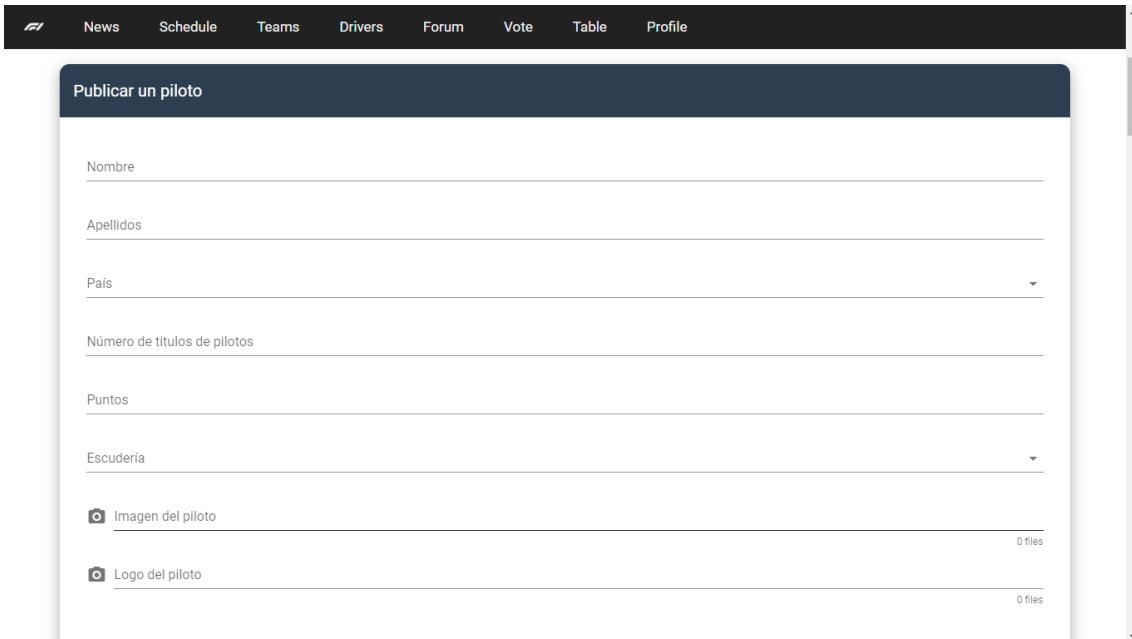
El usuario podrá seleccionar un piloto para visualizar más contenido acerca de él.

8.5.2. Gestión de las pilotos

Para los usuarios que tengan permisos de administrador en la aplicación visualizarán los botones “CREAR PILOTO” y “EDITAR PILOTO”, y dispondrán de la opción de crear, editar o eliminar pilotos en la aplicación.

Para que un usuario con permisos de administrador publique un piloto debe seleccionar la opción “CREAR PILOTO”. La aplicación desplegará un formulario con los datos que se deben rellenar como se muestra en la *Figura 36*.

Aplicación web de fórmula 1



The screenshot shows a web application interface with a dark navigation bar at the top containing the F1 logo and menu items: News, Schedule, Teams, Drivers, Forum, Vote, Table, and Profile. Below the navigation bar is a form titled "Publicar un piloto" (Publish a pilot) with a dark header. The form contains several input fields: "Nombre" (Name), "Apellidos" (Surnames), "País" (Country) with a dropdown arrow, "Número de títulos de pilotos" (Number of pilot titles), "Puntos" (Points), "Escudería" (Team) with a dropdown arrow, "Imagen del piloto" (Pilot image) with a camera icon and "0 files" below it, and "Logo del piloto" (Pilot logo) with a camera icon and "0 files" below it.

Figura 36: Publicar piloto

Para que un usuario con permisos de administrador edite o elimine algún piloto que hay creada debe seleccionar la opción "EDITAR PILOTO". La aplicación lo redireccionará a la página de editar pilotos como se muestra en la *Figura 37*.

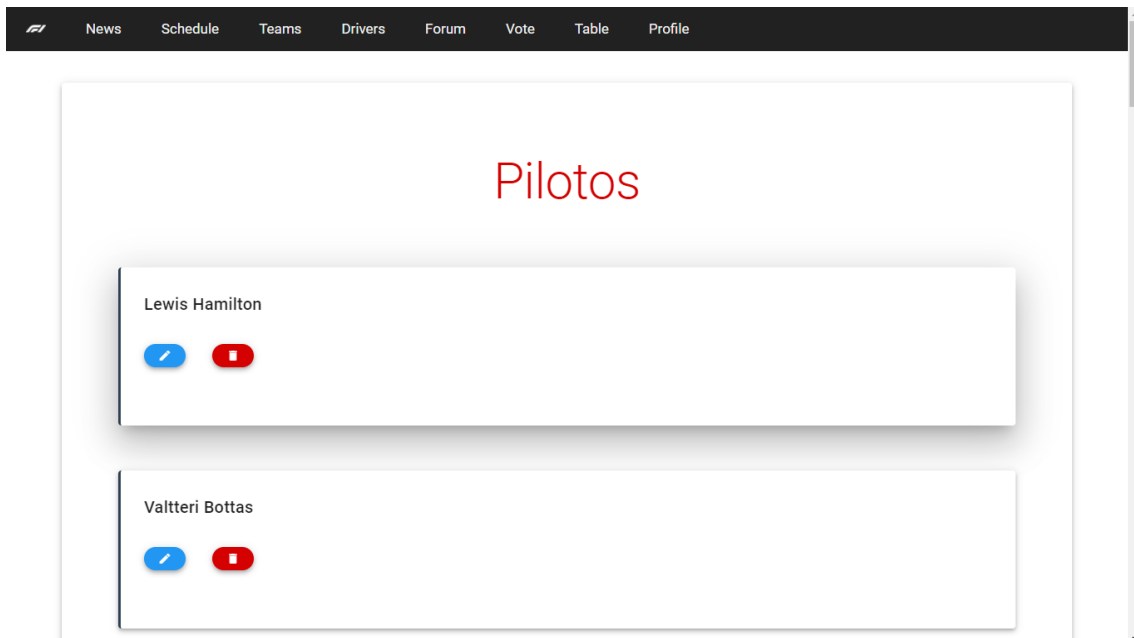




Figura 37: Pilotos

Aplicación web de fórmula 1

Para que un usuario con permisos de administrador edite un piloto debe seleccionar el botón . La aplicación mostrará un formulario con la información actual del piloto y el usuario podrá modificarla y seleccionar “ACTUALIZAR” para confirmar los cambios o seleccionar “CANCELAR” para descartar los cambios.

Para que un usuario elimine un piloto debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.6. Foro

8.6.1. Visualización de los temas

Una vez que el usuario inicie sesión en la aplicación tendrá acceso los temas que hayan creado los usuarios en la aplicación como se muestra en la *Figura 38*, para acceder a los temas el usuario debe seleccionar “Forum” en la barra de navegación.

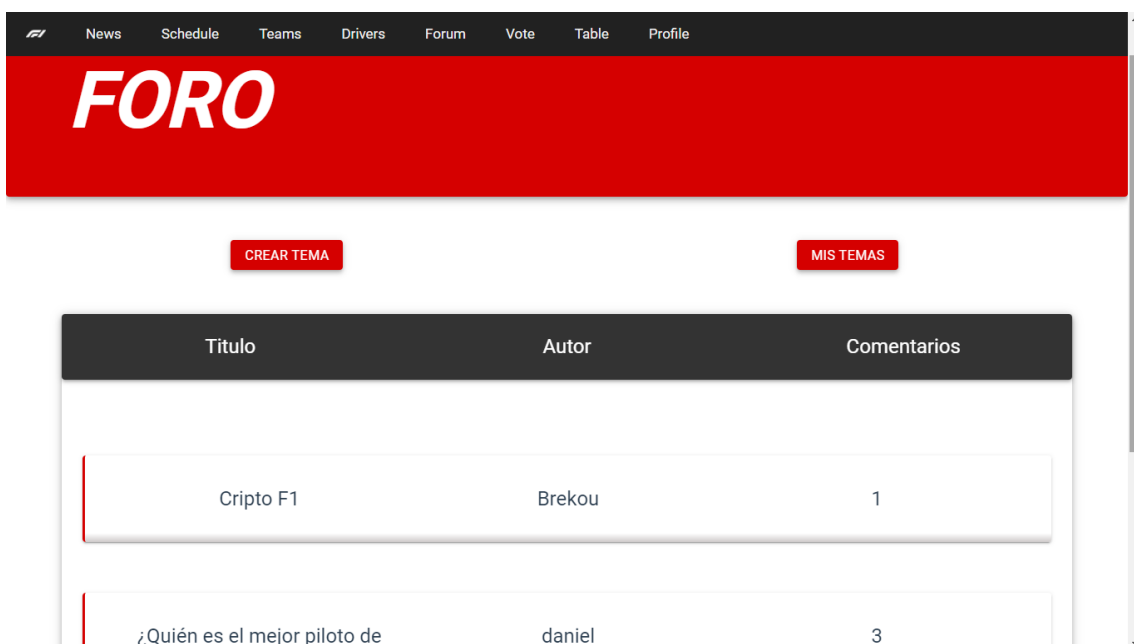


Figura 38: Foro

El usuario podrá seleccionar un tema para visualizar los comentarios que hayan realizado todos los usuarios en ese tema como se muestra en la *Figura 39*.

Aplicación web de fórmula 1



Figura 39: Comentarios de un tema

8.6.2. Gestión de los comentarios de un tema

Para realizar un comentario en un tema el usuario debe seleccionar la opción “COMENTAR”. La aplicación desplegará un formulario similar al mostrado en las noticias como se muestra en la *Figura 40*.

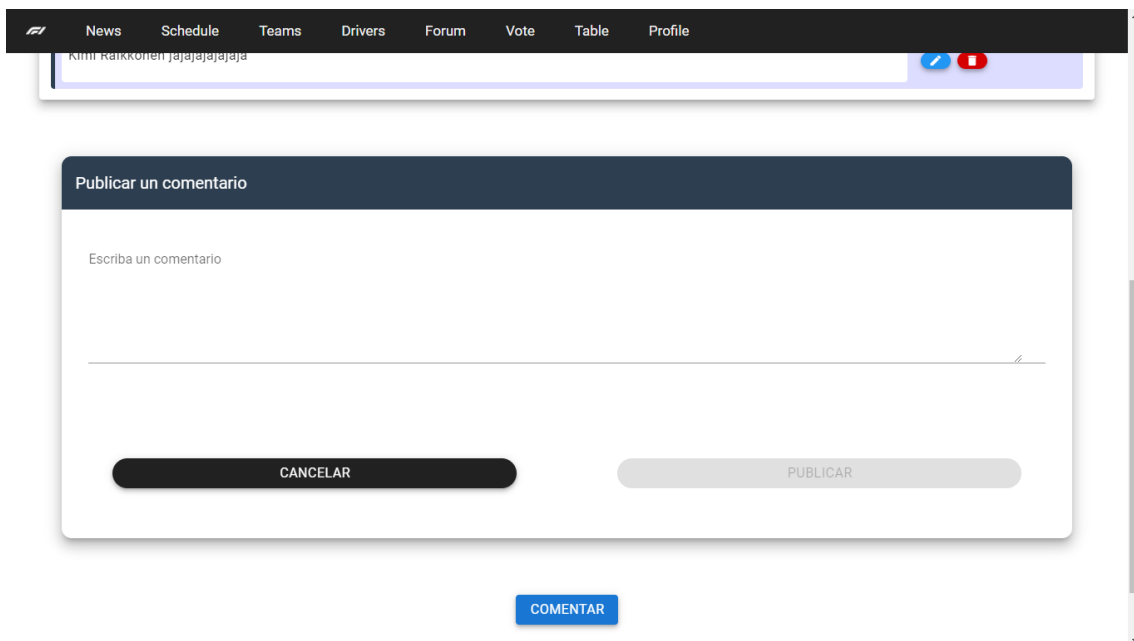




Figura 40: Publicar comentario en un tema

Aplicación web de fórmula 1

Para que un usuario edite cualquier comentario que haya realizado debe seleccionar el botón . La aplicación mostrará el texto actual del comentario y el usuario podrá modificarlo y seleccionar “GUARDAR” para confirmar el cambio o seleccionar “CANCELAR” para descartar el cambio.

Para que un usuario elimine cualquier comentario que haya realizado debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.6.3. Gestión de los temas

Para todos los usuarios de la aplicación visualizarán los botones “CREAR TEMA” y “MIS TEMAS”, y dispondrán de la opción de crear, editar o eliminar temas en la aplicación.

Para que un usuario publique un tema debe seleccionar la opción “CREAR TEMA”. La aplicación desplegará un formulario con los datos que se deben rellenar como se muestra en la *Figura 41*.

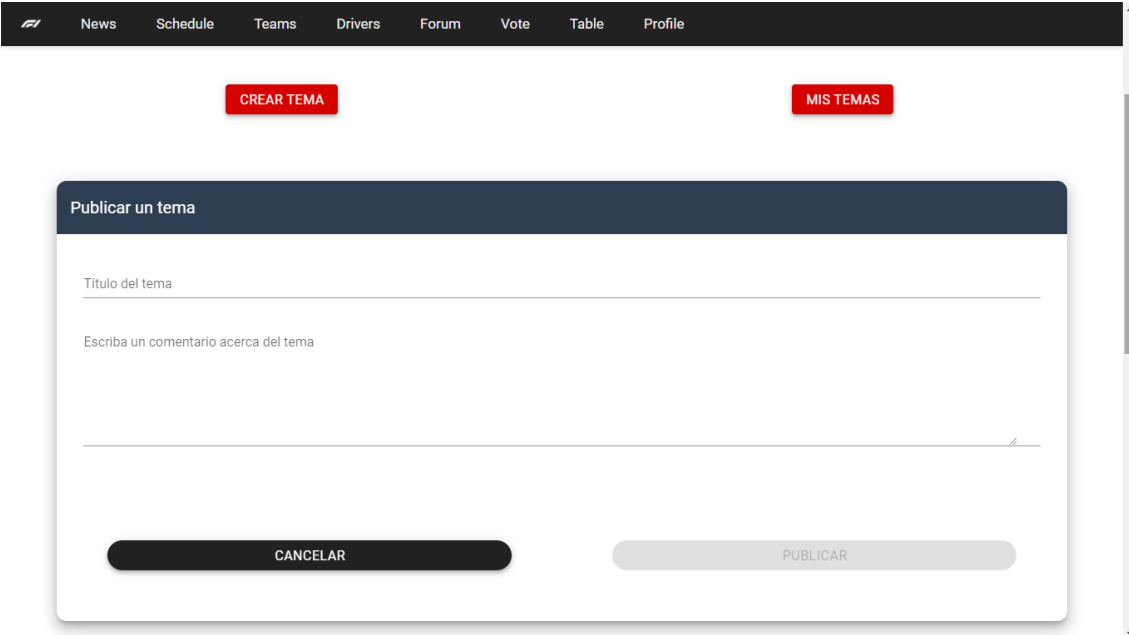



Figura 41: Publicar tema


Para que un usuario edite o elimine algún tema que haya creado debe seleccionar la opción “MIS TEMAS”. La aplicación le redireccionará la página de mis temas como se muestra en la *Figura 42*.

Aplicación web de fórmula 1



Figura 42: Mis temas

Para que un usuario edite un tema debe seleccionar el botón . La aplicación mostrará un formulario con la información actual del tema y el usuario podrá modificarla y seleccionar “ACTUALIZAR” para confirmar los cambios o seleccionar “CANCELAR” para descartar los cambios.

Para que un usuario elimine un tema debe seleccionar el botón . La aplicación mostrará el cuadro de dialogo pidiendo la confirmación al usuario.

8.7. Votaciones

8.7.1. Visualización de las votaciones

Una vez que el usuario inicie sesión en la aplicación tendrá acceso las votaciones, para acceder a ellas el usuario debe seleccionar “Vote” en la barra de navegación como se muestra en la *Figura 43*.

En caso de que haya una votación activa para poder votar el usuario verá la opción “VOTAR” y debe seleccionarla para realizar un voto. La aplicación desplegará un panel con todos los pilotos para seleccionar uno como se muestra en la *Figura 44*. Una vez seleccionado un piloto la aplicación pedirá la confirmación al usuario.

Aplicación web de fórmula 1



Figura 43: Votaciones - votación activa

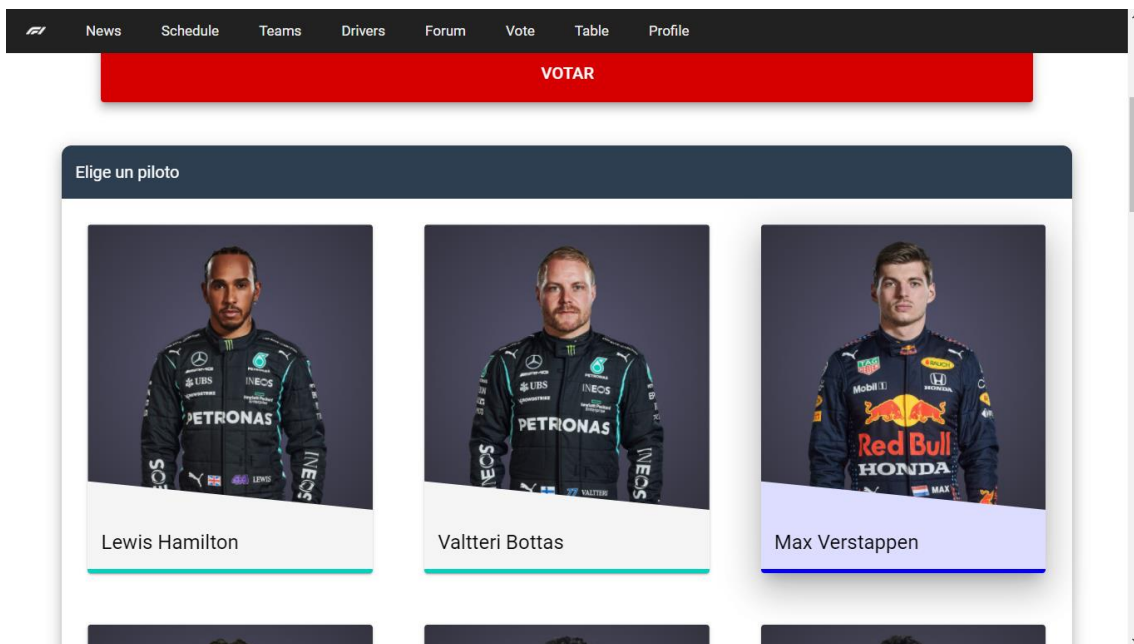


Figura 44: Votar

Para poder ver las votaciones realizadas o cual está activa el usuario debe seleccionar la opción "VER VOTACIONES". La aplicación redireccionará al usuario a una página donde podrá ver las votaciones que se han realizado y si hubiera alguna activa en color azul como se muestra en la *Figura 45*.

Aplicación web de fórmula 1



Figura 45: Ver votaciones

Para poder ver los resultados de una votación finalizada el usuario debe seleccionar una votación. La aplicación redireccionará al usuario a una página donde podrá ver los resultados de los tres pilotos más votados.

Para poder abrir o cerrar una votación solo los usuarios con permisos de administrador podrán realizarlo, en caso de que haya una votación activa la aplicación muestra la opción de cerrar la votación y en el caso de que no haya ninguna dará la opción de abrir una votación nueva como se muestra en la *Figura 46*.



Figura 46: Cerrar o abrir votación

En el caso de querer abrir una votación el usuario con permisos de administrador debe seleccionar "ABRIR VOTACION". La aplicación desplegará un formulario para que el usuario introduzca el nombre del Gran Premio de la votación.

8.8. Clasificación

8.8.1. Visualización de las clasificaciones

Aplicación web de fórmula 1

Una vez que el usuario inicie sesión en la aplicación tendrá acceso la clasificación de escuderías para acceder a ellas el usuario debe seleccionar “Table” en la barra de navegación como se muestra en la *Figura 47*.



Figura 47: Clasificación de escuderías

Si el usuario quiere ver la clasificación de pilotos debe seleccionar “PILOTOS”.

8.8.2. Actualización de las clasificaciones

Para actualizar la clasificación de escuderías y pilotos los usuarios con permisos de administrador deben seleccionar la opción “ACTUALIZAR” como se muestra en la *Figura 48*. La aplicación consultará si hay alguna clasificación más actualizada y almacenará los nuevos resultados para mostrarlos a los usuarios.



Figura 48: Actualizar clasificación

8.9. Perfil

8.9.1. Visualización del perfil

Una vez que el usuario inicie sesión en la aplicación tendrá acceso su perfil para acceder a él debe seleccionar “Profile” en la barra de navegación como se muestra en la *Figura 49*.

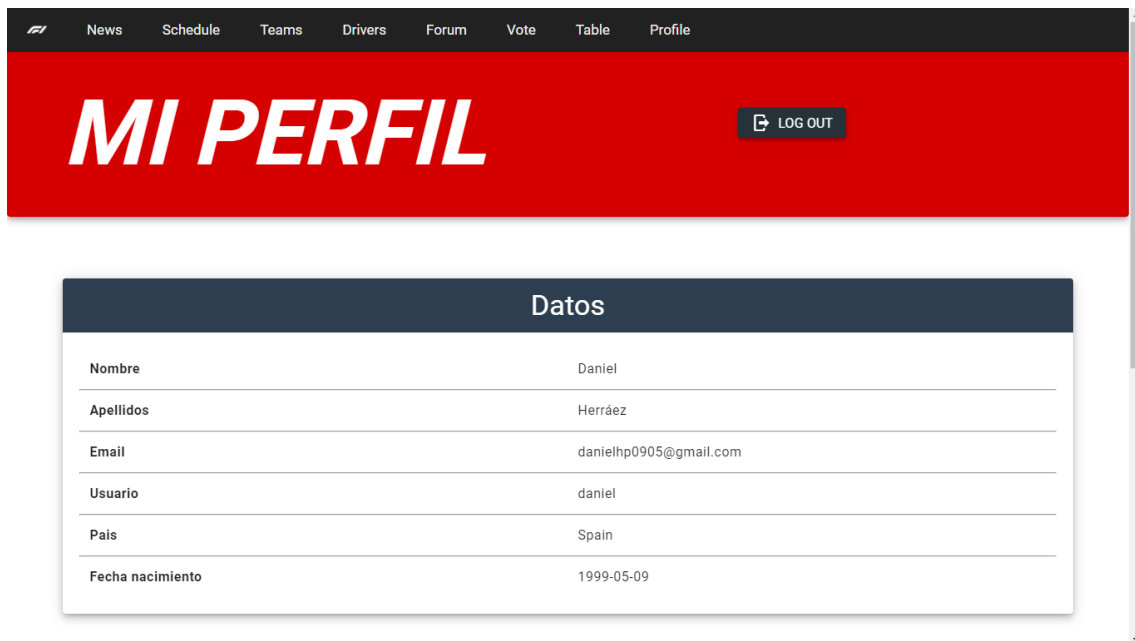


Figura 49: Perfil

Para cerrar la sesión en la aplicación el usuario debe seleccionar “LOG OUT”

8.9.2. Configuración del perfil

En la misma ventana donde el usuario puede visualizar los datos de su cuenta, tiene la opción de modificar algún dato de su cuenta, su contraseña o su email como se muestra en la *Figura 50*. Para ello debe seleccionar una de las tres opciones “PERFIL”, “CONTRASEÑA” o “EMAIL”. La aplicación le mostrará un formulario para cada opción y el usuario deberá rellenarlo con los datos solicitados.

Aplicación web de fórmula 1



Figura 50: Configuración del perfil

8.9.3. Cambiar permisos

Los usuarios con permisos de administrador tienen la opción de dar esos permisos a otros usuarios, para ello el usuario con permisos de administrador debe seleccionar “PERMISOS” como se muestra en la *Figura 51*. La aplicación desplegará un formulario para poder seleccionar el usuario sobre el que se quiere modificar los permisos como se muestra en la *Figura 52*.

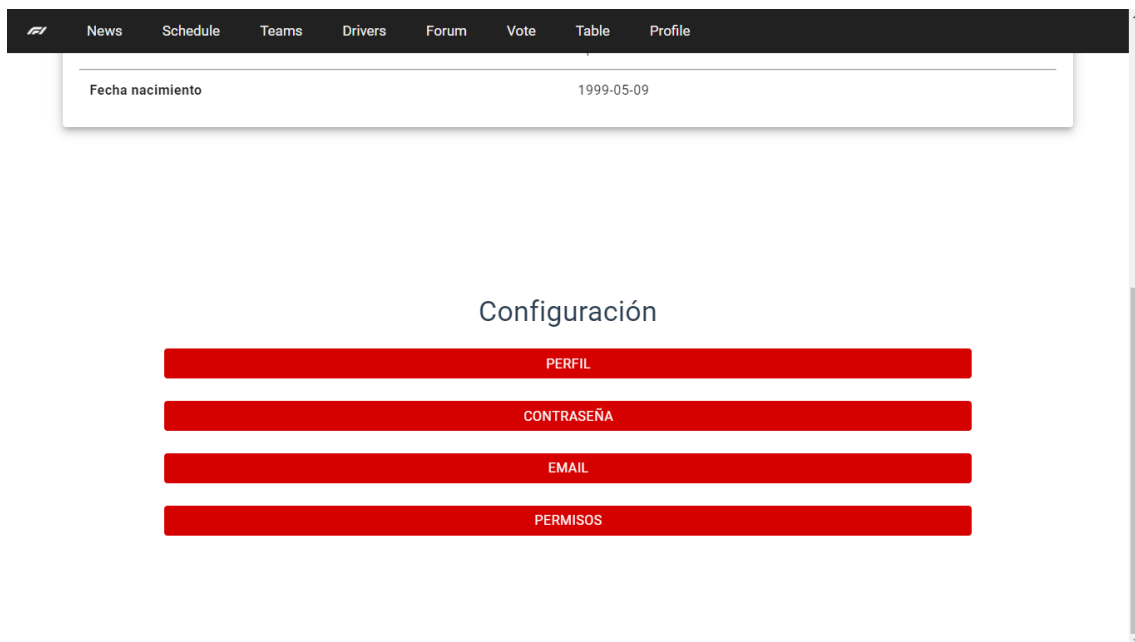


Figura 51: Configuración perfil - Usuario administrador

Aplicación web de fórmula 1

The image shows a web application interface with a dark navigation bar at the top containing the following menu items: News, Schedule, Teams, Drivers, Forum, Vote, Table, and Profile. Below the navigation bar is a modal window titled "Actualizar permisos" (Update permissions). The modal contains the following elements:

- A dropdown menu labeled "Nombre de usuario" (User name).
- A text input field labeled "Contraseña actual" (Current password) with a clear icon on the right.
- Two radio buttons for selecting the permission status:
 - Activar permisos (Activate permissions)
 - Desactivar permisos (Deactivate permissions)
- Two buttons at the bottom: "CANCELAR" (Cancel) and "ACTUALIZAR" (Update).

Figura 52: Cambiar permisos

9. Conclusiones y líneas de trabajo futuras

En este apartado se exponen los objetivos conseguidos durante el desarrollo del proyecto y futuras mejoras a implementar en el mismo.

9.1. Conclusiones

De los objetivos del sistema propuestos al comienzo de proyecto y citados al inicio de este documento:

- Se ha conseguido implementar un sistema de gestión de usuarios para manejar el registro, el inicio y el cierre de sesiones. Además, el sistema permite a los usuarios modificar su información y controlar el acceso de los usuarios a las distintas funcionalidades en función de sus privilegios.
- Se ha conseguido desarrollar una base de datos que permita almacenar toda la información necesaria para mostrar a los usuarios sobre las noticias, el calendario, las escuderías, los pilotos, los temas, los comentarios, las votaciones y sus resultados.
- Se ha implementado la gestión de los comentarios en las noticias y su visualización para cualquier usuario identificado en el sistema. Además, para los usuarios con privilegios de administrador crear, editar y eliminar noticias en el sistema.
- Se ha implementado la gestión de los temas y los comentarios de ellos para cualquier usuario identificado en el sistema.
- Se ha implementado la gestión de las escuderías para los usuarios con permisos de administrador. Además, permitir la visualización de las escuderías para cualquier usuario identificado en el sistema.
- Se ha conseguido desarrollar votaciones sobre los pilotos, calcular los resultados de las mismas y almacenar un histórico con los tres pilotos más votados en cada votación.
- Se ha implementado un sistema de actualización de la clasificación automático cuando un usuario con privilegios lo solicite. Además, de visualizar la información acerca de las clasificaciones.
- Se ha implementado la gestión del calendario de carreras para los usuarios con permisos de administrador. Además, permitir la visualización del calendario para cualquier usuario identificado en el sistema.
- Se ha implementado la gestión de los pilotos para los usuarios con permisos de administrador. Además, permitir la visualización de los pilotos para cualquier usuario identificado en el sistema.
- Mediante el empleo del Material Design se ha conseguido una aplicación con buen nivel de usabilidad. Realizando un diseño de interfaces de usuario responsivo se ha conseguido una alta compatibilidad de la aplicación entre dispositivos.

Aplicación web de fórmula 1

Cabe destacar que los objetivos personales también se han conseguido todos, como el hecho de hacer frente a un proyecto desde el inicio hasta la elaboración de un producto final completamente funcional. Además, se han adquirido conocimientos relacionados con el desarrollo web, empleo de nuevos lenguajes como Vue.js o JavaScript y nuevas herramientas como Firebase.

Otro punto importante es que gracias al desarrollo de este proyecto he podido recordar y emplear conocimientos adquiridos durante los cuatro años; sobre ingeniería del software, planificación de proyectos o programación orientada a objetos entre otros.

9.2. Líneas de trabajo futuras

En este proyecto las posibles mejoras a implementar en un futuro son:

- Permitir el registro e inicio de sesión a través de Google, Facebook o proveedores de servicio similares.
- Empleo de un sistema de caché que almacene la información de la aplicación en el navegador y actualice solamente la necesaria.
- Permitir a los usuarios editar las imágenes de las noticias, el calendario, las escuderías o los pilotos.
- Permitir a los usuarios descargar la aplicación en sus dispositivos móviles según accedan a ella a través de un navegador web.
- Cambiar a una base de datos más comercial como MongoDB [21].

10. Bibliografía

- [1] «FullStack Firebase,» [En línea]. Available: <https://www.fullstackfirebase.com/introduction/what-is-serverless>.
- [2] «JetBrains,» [En línea]. Available: <https://www.jetbrains.com/es-es/webstorm/>.
- [3] «Vue.js,» [En línea]. Available: <https://vuejs.org/>.
- [4] «Vuetify.js,» [En línea]. Available: <https://vuetifyjs.com/en/>.
- [5] «Firebase,» [En línea]. Available: <https://firebase.google.com/docs/firestore>.
- [6] «Vuese,» [En línea]. Available: <https://vuese.org/>.
- [7] «JSDoc,» [En línea]. Available: <https://jsdoc.app/>.
- [8] «Axios,» [En línea]. Available: <https://www.npmjs.com/package/axios>.
- [9] «HTML,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/HTML>.
- [10] «CSS,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/CSS>.
- [11] «JavaScript,» [En línea]. Available: <https://developer.mozilla.org/es/docs/Web/JavaScript>.
- [12] «JSON,» [En línea]. Available: <https://www.json.org/json-es.html>.
- [13] «Node.js,» [En línea]. Available: <https://nodejs.org/es/>.
- [14] «npm,» [En línea]. Available: <https://www.npmjs.com/>.
- [15] «Babel,» [En línea]. Available: <https://babeljs.io/>.
- [16] «ESLint,» [En línea]. Available: <https://eslint.org/>.
- [17] «GitHub,» [En línea]. Available: <https://github.com/>.
- [18] «Visual paradigm,» [En línea]. Available: <https://www.visual-paradigm.com/>.
- [19] «Microsoft Project,» [En línea]. Available: <https://www.microsoft.com/es-es/microsoft-365/project/project-management-software?tab=tabs-1?tab=tabs-1>.
- [20] M. N. Moreno García, "Gestión de Proyectos, Práctica 1 - Estimación del esfuerzo".
- [21] «MongoDB,» [En línea]. Available: <https://www.mongodb.com/es>.