



UNIVERSITAT DE  
BARCELONA

**Trabajo final de grado**

**GRADO DE INFORMÁTICA**

**Facultad de Matemáticas e Informática**

**Universidad de Barcelona**

---

**F1 Data Analysis**

---

**Autor: Arturo Latorre**

**Director: Dra. Anna Puig**

**Realizado en: Departamento de Matemáticas e Informática**

**Barcelona, 14 de enero de 2025**

## Abstract

This Bachelor's Thesis (TFG) aims to analyze data from Formula 1 races in recent years. To achieve this, the analysis, design, and development of an application will be carried out to provide an interactive and engaging experience for motorsport enthusiasts, as well as a valuable resource for both Formula 1 fans and professionals who wish to study the variability and correlation of the various variables involved in the championships. The project has several main objectives. First, it seeks to collect and manage both historical and real-time data from Formula 1 races, covering different types of information such as drivers, teams, circuits, race results, among other relevant aspects.

Additionally, this project develops visualizations that enable users to explore different dimensions of this data, such as performance charts for drivers and teams, race strategy analyses, performance comparisons by circuits, and visualizations of pit stops. An innovative component of this project is the integration of a chatbot based on NLP (Natural Language Processing), which provides support for users to visualize this data. The user interface has been designed to be intuitive and accessible, facilitating navigation and access to the various modules of the application, including the different data visualizations.

## Resum

Aquest Treball de Final de Grau té com a objectiu l'anàlisi de dades procedents de les carreres de la Fórmula 1 en els últims anys. Per això, es farà l'anàlisi, disseny i desenvolupament d'una aplicació per proporcionar una experiència interactiva i atractiva pels amants de l'esport de motor i com un recurs valuós tant com pels entusiastes de la Fórmula 1 com pels professionals que vulguin estudiar la variabilitat i la correlació de diferents variables implicades en els campionats. El projecte té diversos objectius principals. Primer, es vol recopilar i gestionar dades històriques i en temps real de les carreres de la Fórmula 1 incloent diferents tipus d'informació com pilots, equips, circuits, resultats de carreres com altres aspectes rellevants.

A més, en aquest projecte, se'n desenvoluparan diferents visualitzacions que permeten explorar diferents dimensions d'aquestes dades com gràfics de rendiments de pilots i equips, anàlisis d'estratègies de carrera, comparatives de rendiment per circuits i visualitzacions de parades de boxes. Un component innovador dins d'aquest projecte, és la integració d'un xatbot basat en NLP (Natural Language Processing) que permet donar suport als usuaris per visualitzar aquestes dades. La interfície d'usuari s'ha dissenyat per ser intuïtiva i accessible facilitant la navegació i l'accés als diferents mòduls de l'aplicació incloent les diferents visualitzacions de les dades.

## Resumen

Este Trabajo de Fin de Grado (TFG) tiene como objetivo el análisis de datos procedentes de las carreras de Fórmula 1 en los últimos años. Para ello se realizará el análisis, diseño y desarrollo de una aplicación para proporcionar una experiencia interactiva y atractiva para los amantes del deporte de motor y como un recurso valioso tanto para los entusiastas de la Fórmula 1 como para los profesionales que desean estudiar la variabilidad y la correlación de las distintas variables implicadas en los campeonatos. El proyecto tiene varios objetivos principales. Primero, se pretende recopilar y gestionar datos históricos y en tiempo real de las carreras de la Fórmula 1 abarcando diferentes tipos de información como los pilotos, los equipos, los circuitos, resultados de carreras entre otros aspectos relevantes.

Además, en este proyecto se desarrollan visualizaciones que permiten explorar diferentes dimensiones de estos datos como gráficos de rendimiento de pilotos y equipos, análisis de estrategias de carrera, comparativas de rendimiento por circuitos y visualización de paradas de boxes. Un componente innovador dentro de este proyecto es la integración de un chatbot basado en NLP (Natural Language Processing) que permite dar soporte a los usuarios para visualizar dichos datos. La interfaz de usuario se ha diseñado para ser intuitiva y accesible facilitando la navegación y el acceso a los diferentes módulos de la aplicación incluyendo las diferentes visualizaciones de los datos.

## **Agradecimientos**

En primer lugar, quiero dar agradecimientos a mi tutora; Dra. Anna Puig por su orientación y el apoyo durante todo el desarrollo de este proyecto y la gran oportunidad de realizar este proyecto.

En segundo lugar, a mi madre, Caterina Castelltort, por su incondicional apoyo a lo largo de esta etapa de mi vida y por estar siempre a mi lado cuando más lo he necesitado.

Por último, pero no menos importante, quiero extender mi agradecimiento a toda mi familia, a mi pareja y a mis amigos, quienes han sido un pilar fundamental con su apoyo incondicional en todo momento.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	2
1.3. Planificación . . . . .	3
1.4. Organización del documento . . . . .	4
<b>2. Antecedentes</b>	<b>5</b>
2.1. Contexto . . . . .	5
2.1.1. Historia . . . . .	5
2.1.2. Campeonatos . . . . .	7
2.1.3. Carreras . . . . .	8
2.1.4. Pilotos, escuderías y vehículos . . . . .	9
2.2. Requerimientos . . . . .	13
2.2.1. Hipótesis sobre los datos . . . . .	13
2.2.2. Requerimientos Funcionales . . . . .	14
2.2.3. Requerimientos No Funcionales . . . . .	14
2.3. Datos de la Fórmula 1 . . . . .	15
2.4. Chatbots para Visualización de Datos . . . . .	16
2.4.1. Estado actual de las aplicaciones de la Fórmula 1 . . . . .	17
2.4.2. Retrieval Augmented Generation . . . . .	18
2.5. Conclusiones . . . . .	20
<b>3. Análisis</b>	<b>21</b>
3.1. Casos de Uso . . . . .	21
3.1.1. Perfiles de Usuario . . . . .	21
3.1.2. Casos de Uso . . . . .	23
3.2. Diseño gráfico y story board . . . . .	26

3.2.1.	UC1. Mostrar todos los pilotos . . . . .	27
3.2.2.	UC2: Mostrar todos los constructores . . . . .	27
3.2.3.	UC3: Mostrar todas las carreras . . . . .	29
3.2.4.	UC4: Mostrar las tablas de clasificación de los pilotos . . . . .	32
3.2.5.	UC5: Mostrar las tablas de clasificación de los constructores . . . . .	33
3.2.6.	UC6: Analizar correlaciones entre datos . . . . .	34
3.2.7.	UC7: Hablar con el chatbot . . . . .	35
3.3.	Propiedades del Chatbot . . . . .	37
3.3.1.	Propiedades del Chatbot . . . . .	37
3.3.2.	Impacto en la Aplicación . . . . .	38
3.4.	Datos . . . . .	38
<b>4.</b>	<b>Diseño</b>	<b>39</b>
4.1.	Arquitectura general de la aplicación . . . . .	39
4.2.	Modelo . . . . .	40
4.2.1.	Diseño del Chatbot . . . . .	40
4.2.2.	Diseño de la Base De Datos . . . . .	42
4.3.	View y ViewModel . . . . .	43
4.3.1.	View . . . . .	43
4.3.2.	ViewModel . . . . .	48
4.4.	Desarrollo de la aplicación . . . . .	49
4.4.1.	Arquitectura de la aplicación desarrollada . . . . .	49
4.4.2.	Backend . . . . .	50
<b>5.</b>	<b>Resultados</b>	<b>57</b>
5.1.	Resultados Obtenidos . . . . .	58
5.1.1.	Usuario Casual: Visualizaciones . . . . .	58
5.1.2.	Usuario Experto: Visualizaciones . . . . .	77
5.2.	Experiencia preliminar con usuarios . . . . .	81
<b>6.</b>	<b>Conclusiones y líneas futuras</b>	<b>83</b>
<b>A.</b>	<b>Manual Técnico</b>	<b>88</b>
A.1.	Estructura del Proyecto . . . . .	88
A.2.	Ajustes en Archivos . . . . .	90
A.2.1.	Pilotos y Calendario Carreras . . . . .	90

A.2.2. Constructores . . . . . 90

A.3. Como poblar la base de datos . . . . . 91

# Capítulo 1

## Introducción

En este capítulo, se introduce el ámbito del proyecto, explicando su origen y su necesidad. A continuación, se detallan los objetivos que se pretenden alcanzar y la planificación llevada a cabo para estructurar el desarrollo del proyecto. Finalmente, se presenta la organización de este documento.

### 1.1. Motivación

La Fórmula 1 es un deporte que genera una cantidad masiva de datos en cada carrera: información de tiempos de vuelta, posiciones, estrategias, pit stops, entre otros datos. Sin embargo, gran parte de estos datos no son accesibles para los aficionados de la Fórmula 1 de manera fácil, ya que son de difícil acceso para poder hacer un análisis exhaustivo de los mismos. Por ello, surgió la necesidad de realizar una aplicación que permitiera no solo recopilar y organizar estos datos de forma sistemática, sino también poder analizarlos y visualizarlos de forma fácil.

En general, el análisis visual de los datos facilita la comprensión de las relaciones entre las distintas variables a analizar, pero a menudo, cuando se están analizando datos masivos, el usuario puede perderse en su exploración. Así, un segundo problema que surge es el poder ayudar al usuario en el análisis. La reciente aparición de LLM ofrece una oportunidad de solucionar o mitigar este problema. En este aspecto, la integración de un chatbot a través de consultas de los datos y las visualizaciones mediante lenguaje natural facilitarían esta exploración.

Así pues, el alcance de este proyecto se enmarca en la visualización y análisis de datos, con especial énfasis en la generación de visualizaciones dinámicas e interactivas, ofreciendo a la vez explicaciones claras y contextualizadas en cada momento. La integración de datos en tiempo real y la inclusión de toda la telemetría disponible son puntos que quedan fuera del alcance del presente proyecto.

## 1.2. Objetivos

Este proyecto tiene como objetivo general desarrollar una aplicación interactiva para la visualización y análisis de datos históricos de la Fórmula 1. Esta aplicación debe permitir a los usuarios acceder, comprender y analizar información como tiempos de vuelta, posiciones y pit stops mediante visualizaciones y un chatbot integrado que responde a consultas de lenguaje natural, tanto sobre los datos, como sobre las visualizaciones.

El objetivo general se desglosa en los siguientes objetivos específicos:

- **Análisis, diseño e implementación de la arquitectura de la aplicación:** La aplicación a desarrollar tiene que ser capaz de adaptarse a futuras ampliaciones, asegurando la escalabilidad. Para ello será necesario diseñar una arquitectura concreta con la elección de las herramientas software necesarias para garantizar un desarrollo escalable y mantenible.
- **Diseñar e implementar una base de datos:** Se pretende realizar una base de datos que almacene y organice de manera óptima los datos de la Fórmula 1.
- **Diseñar y desarrollar gráficas dinámicas:** Se desea hacer una aplicación que permita generar gráficas dinámicas, interactivas y personalizadas basadas en los datos como correlaciones entre variables.
- **Diseño, desarrollo e integración de un chatbot:** Facilitar la interacción con los datos y las visualizaciones mediante consultas en lenguaje natural, permitiendo a los usuarios obtener explicaciones claras y generar visualizaciones según estas.
- **Garantizar la accesibilidad a diferentes usuarios:** Se quiere atraer a todo tipo de usuario, desde usuarios casuales como a usuarios expertos en el análisis de datos.

### 1.3. Planificación

El proyecto se ha estructurado en distintos Sprints como se puede ver en la Figura 1.1, organizados por cada 16 días, donde se realizarán las siguientes tareas para los objetivos comentados en la Sección anterior:

Sprints	Sprint1	Sprint2	Sprint3	Sprint4	Sprint5	Sprint6
	10 Sept - 01 Oct	01 Oct- 22 Oct	22 Oct- 12 Nov	12 Nov - 03 Dic	03 Dic - 24 Dic	24 Dic- 15 Ene
Investigación sobre chatbots						
Investigación sobre los datos de la Fórmula 1						
Creación de la base de datos						
Desarrollo del diseño						
Desarrollo de las gráficas						
Implementación del chatbot						
Mejoras en la aplicación						
Redacción de la memoria						
Avaluación de Usuarios						
Revisión de la Memoria						

Figura 1.1: Gráfica de Planificación, organizada por Sprints

## 1.4. Organización del documento

Esta memoria se desglosa en los siguientes capítulos:

- **Antecedentes:** Se presenta el contexto de la Fórmula 1, abarcando la historia, como campeonatos, carreras, pilotos, y escuderías. Además, se analizan las hipótesis, los requerimientos del proyecto, las herramientas para la recopilación y visualización de datos.
- **Análisis:** Se identifican los diferentes usuarios y los casos de uso asociados. También se detalla el storyboard mediante el diseño de cada funcionalidad. Además, se examinan los requisitos del chatbot y de los datos necesarios para el proyecto.
- **Diseño:** El diseño y desarrollo de la aplicación se interpreta con la arquitectura general del sistema. Detallando la estructura de la base de datos, el diseño del chatbot, la interacción entre visualización y lógica, y las características del backend.
- **Resultados:** Se analizan los resultados de cada caso de uso. Además, se realiza una experiencia preliminar con usuarios para evaluar el funcionamiento de la aplicación.
- **Conclusiones:** Se redactan la resolución de los objetivos alcanzados y se analizan las áreas de mejora.
- **Anexo:** Se incluye un manual técnico que detalla la estructura del proyecto, y la forma de poblar la base de datos.

# Capítulo 2

## Antecedentes

En este capítulo, se presenta el contexto de la aplicación, ofreciendo una visión general de los elementos clave que conforman la Fórmula 1, incluyendo su historia, campeonatos, carreras como los pilotos, las escuderías y algunos componentes de los vehículos. También se detallan las hipótesis y los requerimientos, orientados para satisfacer las necesidades del usuario. Además, se analiza el uso de los datos de la Fórmula 1, destacando las principales API disponibles para la recopilación y la gestión. Finalmente, se exploran las herramientas de visualización y análisis basadas en chatbots ya existentes, evaluando sus capacidades a través de un enfoque comparativo, y se concluye con un análisis sobre la posible integración de estas tecnologías en una aplicación orientada a analizar los datos de la Fórmula 1 en los últimos años.

### 2.1. Contexto

#### 2.1.1. Historia

Las carreras han estado presentes desde la invención de los automóviles, reflejando el espíritu de la competencia e innovación. Las primeras pruebas automovilísticas comenzaron en Francia en el 1884, cuando las carreras se llevaban a cabo entre pueblos en caminos de tierra. A medida que la automoción avanzaba las competiciones se volvieron más rápidas y sofisticadas, lo que las hacía más emocionantes, pero también más peligrosas debido a la falta de medidas de seguridad en estos eventos [1]. No fue hasta 1900 que se organizó el primer evento automovilístico de manera más formal, conocido como Gordon-Bennet Cup o la Copa Gordon Bennet [2], una competición anual que tuvo lugar entre 1900 y 1905. Esta permitía a cada país inscribir hasta tres vehículos, estableciendo un precedente en la organización de competiciones internacionales, y atrayendo la atención de varias compañías automovilísticas.

La primera carrera conocida como Gran Prix o Gran Premio se llevó a cabo en 1906 en el pueblo de Le Mans (ver Figura 2.1), organizada por el Automobile Club de France [3]. Esta competición se desarrolló durante dos días debido a la gran longitud del circuito, con vueltas de aproximadamente 105 kilómetros cada una, para un total de 12 vueltas.

En esta histórica carrera participaron un total de 32 competidores, representando a 12 fabricantes diferentes, entre ellos Renault, Mercedes y Fiat, marcas que hoy en día son reconocidas internacionalmente. Aunque este evento no está directamente relacionado con las actuales 24 Horas de Le Mans, sentó las bases para el desarrollo de competiciones de resistencia que hoy en día son mundialmente conocidas y consideradas entre las más prestigiosas del automovilismo. Además, en estos primeros años, surgieron otras competiciones destacadas como la Targa Florio y la carrera de Dieppe, que contribuyeron al crecimiento del deporte. Más adelante, en el 1909 se inauguró el Indianápolis Motor Speedway y en 1922, el Autódromo Nazionale Monza, circuitos icónicos que han dejado huella en la historia.



Figura 2.1: Ferenc Szisz y su mecánico Marteau en Le Mans 1906. Foto extraída de Almacén F1.

En 1904 se formó la AIACR (Association Internationale des Automobile Clubs Reconnus) en París con la misión de regular de forma oficial cada Gran Premio. Entre sus funciones más destacadas se encontraba garantizar que cada competición siguiera una normativa coherente y asegurar una equidad entre los participantes. Además, esta asociación comenzó a sentar las bases para las regulaciones técnicas que con el tiempo, serían claves para la seguridad del deporte, aunque en sus inicios no era una de sus principales prioridades. La AIACR creó dos campeonatos internacionales de monoplazas, el World Manufacturers Championship y la European Grand Prix Championship, pero estos fueron suspendidos en el 1939 por la Segunda Guerra Mundial. A pesar de la suspensión de ambos campeonatos, las pruebas fueron creciendo de cinco a tener dieciocho en 1934, sin embargo, únicamente nombraron cuatro como Gran Prix.

En 1947, dos años después del final de la Segunda Guerra Mundial, la asociación fue reformada y recibió el nombre actual de FIA (Fédération Internationale de l'Automobile). Esta asociación fue la encargada de establecer nuevas regulaciones para las carreras internacionales e introduciendo diferentes divisiones según la cilindrada de los vehículos: Fórmula 1, Fórmula 2, Fórmula 3.

La primera carrera del Mundial de la Fórmula 1 bajo estas nuevas regulaciones se llevó a cabo el 13 de mayo de 1950 en el Circuito de Silverstone. Desde ese momento, la Fórmula 1 ha ido ajustando sus normas y competiciones dependiendo de los sucesivos progresos tecnológicos hasta la actualidad consolidándose como uno de los deportes más prestigiosos y reconocidos a nivel internacional. Actualmente, la Fórmula 1 se estructura en campeonatos anuales formados por distintas carreras en los que los constructores de las distintas escuderías compiten fichando a distintos pilotos. A continuación se detallan las normas de los campeonatos, sus principales componentes y el papel que juegan los pilotos en las escuderías.

### **2.1.2. Campeonatos**

La Fórmula 1 ha cambiado de manera notable desde su primer campeonato en 1950, no solo en aspectos tecnológicos, sino también en la forma en la que se organizan y puntúan las competiciones. Un campeonato consiste en un conjunto de carreras que se llevan a cabo a lo largo del año, empezando a finales de febrero y terminando a finales de noviembre, abarcando entre 22 y 24 carreras, dependiendo de los contratos vigentes, y conocidas como Grandes Premios. En estas competiciones participan 10 escuderías, cada una de las cuales compete con un máximo de dos pilotos.

#### **Calendario**

El calendario de las carreras de Fórmula 1 varía cada temporada, adaptándose a los contratos anuales firmados entre los organizadores de los Grandes Premios y la Fórmula 1. Esto implica que algunos circuitos tradicionales puedan ser remplazados por nuevas sedes dependiendo de diversos factores. A pesar de estos cambios, el calendario siempre busca mantener una representación global en todos o la gran mayoría de continentes.

#### **Formato de carrera**

Cada carrera tiene una distancia programada de 305 kilómetros como mínimo. En caso de no alcanzar el límite, por ejemplo por condiciones climatológicas, se finalizará según la posición en ese momento, y se repartirán los puntos según la distribución descrita en la Tabla 2.1.

Las carreras comienzan con una vuelta de formación, en la que los pilotos calientan los neumáticos y los frenos, además de comprobar el estado del vehículo. Una vez alineado el último vehículo de la parrilla de salida, se dará la bandera verde, y comenzará el apagado de luces del semáforo.

#### **Formato al Sprint**

Desde 2021 se introdujo el formato de carreras al Sprint en ciertos fines de semana

seleccionados, y anunciados a principio de año. Este formato cubre una distancia aproximada de 100 kilómetros en la carrera al Sprint, lo que equivale a un tercio de la carrera principal [4].

- **Formato:** Los fines de semana con este formato, únicamente tienen una única Práctica. Después contaremos con la clasificación para la carrera al Sprint. El sábado se realiza la carrera al Sprint, y por la tarde se realizará la clasificación normal. Y el domingo, se realizará la carrera en formato normal.
- **Sin paradas obligatorias:** A diferencia de las carreras principales, en este formato, no es obligatorio realizar una parada en boxes durante el Sprint.
- **Puntuación:** Actualmente, se reparten únicamente puntos a los primeros 8 pilotos, empezando con 8 puntos para el primer piloto, y de forma decreciente hasta el octavo lugar.

Estas carreras son un formato innovador, donde los pilotos buscan un enfoque más agresivo, y los espectadores viven una experiencia más emocionante debido a la intensidad y la duración de dicho formato.

### 2.1.3. Carreras

Las **carreras** de la Fórmula 1, o *races*, representan uno de los papeles más importantes de todo el fin de semana de la competición. Cada carrera es única y se desarrolla en un circuito concreto con las consecuentes estrategias de los distintos equipos.

#### Clasificación

La clasificación de la carrera [5] es una de las columnas principales de la Fórmula 1, ya que se determina el orden de la parrilla de salida para la carrera principal. El formato actual consta de tres rondas eliminatorias con tiempos diferentes para cada sesión.

- **Q1:** Participan los 20 pilotos en un tiempo de 18 minutos. Los cinco más lentos son eliminados y ocupan las últimas posiciones de la parrilla.
- **Q2:** Participan los 15 pilotos restantes durante 15 minutos. Los cinco más lentos son eliminados y se colocan entre la 11ª y la 15ª posición.
- **Q3:** Los últimos 10 pilotos más rápidos compiten por la pole position en una sesión de 12 minutos.

Este formato asegura que los pilotos más rápidos ocupen las primeras posiciones de salida. Una vez definida la parrilla de salida, el siguiente objetivo de los pilotos es sumar la mayor cantidad de puntos posibles en la carrera.

## Sistema de Puntos

El sistema de puntos ha ido cambiando a lo largo de la historia para premiar de la manera más equitativa a los mejores pilotos y equipos. Desde 2010 no se ha modificado el sistema de puntos actual, donde beneficia a los 10 primeros clasificados de cada carrera, distribuidos tal y como muestra la Tabla 2.1 [6]. En 2022, después del Gran Premio de Bélgica del 2021, se introdujo un sistema de puntos para circunstancias excepcionales en las que una carrera no completada en su totalidad debido a factores externos [7].

Posición	100 %	50 % - 75 %	25 % - 50 %	<25 %
1º lugar	25 puntos	19 puntos	13 puntos	6 puntos
2º lugar	18 puntos	14 puntos	10 puntos	4 puntos
3º lugar	15 puntos	12 puntos	8 puntos	3 puntos
4º lugar	12 puntos	9 puntos	6 puntos	2 puntos
5º lugar	10 puntos	8 puntos	5 puntos	1 punto
6º lugar	8 puntos	6 puntos	4 puntos	-
7º lugar	6 puntos	5 puntos	3 puntos	-
8º lugar	4 puntos	3 puntos	2 puntos	-
9º lugar	2 puntos	2 puntos	1 punto	-
10º lugar	1 punto	1 punto	-	-

Tabla 2.1: Sistema de puntos según el tanto por ciento de la distancia completada. Por ejemplo, 100 % significa que se han completado todas las vueltas de la carrera

Además, el piloto que obtiene la vuelta rápida o fast lap y queda entre los diez primeros pilotos, se le suma 1 punto más. El sistema de puntos actual no solo determina las posiciones en el campeonato, sino que también influye en las estrategias de cada equipo, y ajustan sus decisiones en función de los puntos que puedan obtener en cada situación, buscando siempre maximizar su rendimiento a lo largo de toda la temporada. En el caso de los constructores, sus puntos es la suma total de ambos pilotos que le representan.

### 2.1.4. Pilotos, escuderías y vehículos

En la Fórmula 1, los pilotos y los constructores son los dos pilares más importantes en la competición. Los **pilotos** o *drivers* representan a los **escuderías** o *constructors* en la pista, que son los responsables del diseño del monoplaza, y junto a ellos, elaboran las estrategias para cada carrera.

Desde 2018, con la revolución de AWS, la parrilla ha experimentado cambios significativos, tanto en los nombres de los constructores como en la alineación de los pilotos,

desde Force India a Aston Martin o de Toro Rosso a Visa Cash App RB. Estos cambios reflejan la evolución de la Fórmula 1, donde los acuerdos comerciales, y las estrategias a largo plazo tienen un impacto directo en los pilotos y constructores. En la Tabla 2.2 se muestran las escuderías y sus pilotos en el año 2018 y en el año 2024. Como se puede observar, hay diferentes constructores que han desaparecido o que han cambiado de nombre, así como altas y bajas de pilotos. También se puede ver que algunos han cambiado de equipo, como Fernando Alonso, que estaba en McLaren en el 2018 y, en cambio, actualmente está en Aston Martin en 2024.

2018		2024	
Constructores	Pilotos	Constructores	Pilotos
Red Bull Racing	Max Verstappen	Red Bull Racing	Max Verstappen
	Daniel Ricciardo		Sergio Pérez
Mercedes	Lewis Hamilton	Mercedes	Lewis Hamilton
	Valtteri Bottas		George Russell
Ferrari	Sebastian Vettel	Ferrari	Charles Leclerc
	Kimi Räikkönen		Carlos Sainz
Force India	Sergio Pérez	Aston Martin	Fernando Alonso
	Esteban Ocon		Lance Stroll
Haas F1 Team	Romain Grosjean	Haas F1 Team	Kevin Magnussen
	Kevin Magnussen		Nico Hülkenberg
McLaren	Fernando Alonso	McLaren	Lando Norris
	Stoffel Vandoorne		Oscar Piastri
Renault	Nico Hülkenberg	Alpine	Esteban Ocon
	Carlos Sainz		Pierre Gasly
Sauber	Charles Leclerc	Kick Sauber	Valtteri Bottas
	Marcus Ericsson		Zhou Guanyu
Toro Rosso	Pierre Gasly	Visa Cash App RB	Yuki Tsunoda
	Brendon Hartley		Daniel Ricciardo, Liam Lawson
Williams	Lance Stroll	Williams	Alexander Albon
	Sergey Sirotkin		Logan Sargeant, Franco Colapinto

Tabla 2.2: Tabla comparativa entre constructores y pilotos del año 2018 y 2024.

El desarrollo de los equipos y la llegada de nuevas promesas como Oscar Piastri y George Russell han sido claves para mantener la competitividad en la Fórmula 1. Al mismo tiempo, pilotos más experimentados como Fernando Alonso y Lewis Hamilton

siguen aportando al campeonato con su amplia experiencia, asegurando un equilibrio entre los talentos emergentes y la veteranía.

La evolución constante de la Fórmula 1, no solo se refleja en los cambios de alineaciones y constructores, sino también en la necesidad de adaptarse a un entorno cada vez más impulsado por datos y decisiones estratégicas. En este contexto, se destaca la importancia de nuevas herramientas tecnológicas que permitan analizar toda esta información usando diferentes visualizaciones que ayuden a interpretar los datos.

Los monoplazas cada vez son más modernos, incorporando nuevas tecnologías para la evolución como los neumáticos, diseñados para ofrecer el máximo rendimiento en diferentes condiciones, o los pit stops, donde la coordinación puede cambiar el curso de la carrera. Además, el sistema de reducción de arrastre (DRS) y los sensores de velocidad contribuyen tanto a mejorar la competitividad en pista como a proporcionar datos para el análisis estratégico.

## Neumáticos

El proveedor exclusivo de neumáticos para la Fórmula 1 es Pirelli [8], donde ofrece diferentes compuestos de neumáticos cada temporada, que varían en durabilidad y rendimiento.

En el 2018, existía un abanico más amplio de compuestos para las carreras en seco, ver Figura 2.2. A diferencia del sistema actual, que simplifica los neumáticos en cinco categorías principales para condiciones de seco y dos para mojado, en el 2018 existían hasta siete categorías distintas para seco, además de las dos categorías de mojado que se mantienen en la actualidad, ver Figura 2.3. Este cambio de neumáticos ha permitido una mayor claridad en las estrategias de carreras, manteniendo la esencia competitiva y reduciendo la confusión de los espectadores [9].

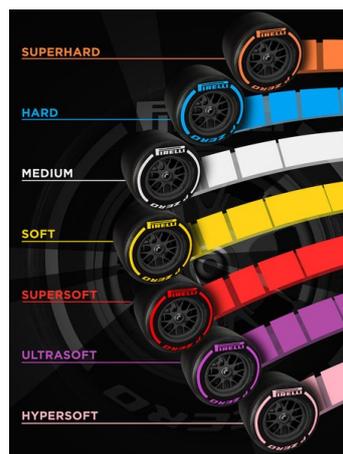


Figura 2.2: Diferentes categorías de neumáticos en seco de la Formula 1 del 2018. Foto modificada extraída de Motorsport.



Figura 2.3: Diferentes categorías de neumáticos de la Formula 1. Foto modificada extraída de Pirelli.

Los compuestos más blandos son los más rápidos, pero tienen una mayor degradación, mientras que los compuestos más duros ofrecen mayor durabilidad pero menor velocidad. En cambios los compuestos Intermedios y de Lluvia extrema únicamente se utilizan en caso de lluvia, y se eligen dependiendo de la cantidad de agua en la pista.

Como norma general, cada piloto debe usar al menos dos compuestos diferentes de neumáticos. Este reglamento obliga a que los equipos planifiquen cuidadosamente sus estrategias, teniendo en cuenta el desgaste de los neumáticos, las condiciones de pista y las posibles paradas en boxes de sus rivales. El rendimiento y la durabilidad de los neumáticos también influyen en la estrategia de los pit stops, donde cada segundo cuenta.

### **Pit Stops**

Los **Pit Stops** [10] son paradas en boxes que se realizan durante el transcurso de la carrera para cambiar los neumáticos del coche que tienen un impacto directo en la estrategia de cada piloto. Un cambio rápido de neumáticos puede marcar la diferencia entre ganar o perder posiciones. Normalmente, el tiempo promedio de un Pit Stop exitoso oscila entre 1.5 segundos a 3 segundos. Sin embargo, si el piloto tiene una penalización de tiempo, esta debe cumplirse antes de realizar el cambio de neumáticos, y durante ese período de penalización, ningún mecánico puede tocar el vehículo.

Antes de entrar al Pit Lane, los vehículos deben cumplir con un límite de velocidad máximo de 80 kilómetros por hora excepto en ciertos Grandes Premios que es de 60 kilómetros por hora, establecido por la FIA para garantizar la seguridad de los mecánicos y los equipos. Exceder este límite conlleva una penalización de tiempo severa [11].

### **Drag Reduction System**

El Drag Reduction System (DRS) un mecanismo que permite a los pilotos abrir el alerón trasero en zonas específicas del circuito, reduciendo así su resistencia aerodinámica, y consiguiendo entre unos 15 y 20 kilómetros por hora adicionales. Este sistema facilita a los pilotos el adelantamiento, pero únicamente se puede activar si el piloto está a menos de un segundo del coche de delante según los datos obtenidos de los sensores de velocidad, y si la FIA da permiso para poder utilizarlo [12].

## **Sensores de Velocidad**

Desde 2018, la Fórmula 1 colabora con Amazon Web Services (AWS) para poder gestionar y procesar en tiempo real los datos masivos generados en cada carrera, permitiendo un análisis inmediato mediante Machine Learning para poder tomar decisiones estratégicas, y brindar estadísticas detalladas para los aficionados, y mejorando la experiencia del espectador. Cada monoplaza está equipado con más de 300 sensores distribuidos estratégicamente, generando hasta 1.1 millones de datos por segundo. Estos sensores tienen diferentes funciones incluyendo la velocidad del monoplaza en diferentes sectores del circuito, además de la zona de detección del DRS o el Pit Lane, como también monitoriza otros parámetros cruciales de cada monoplaza [13].

A continuación se detallan los requerimientos deseables en la obtención de los datos generados por estos campeonatos, así como la interacción con ellos.

## **2.2. Requerimientos**

En la Fórmula 1, la capacidad de analizar datos complejos y presentar datos visuales de manera intuitiva a los espectadores es fundamental para facilitar las tomas de decisiones y enriquecer su experiencia. En este contexto, el análisis de los datos generados por la Fórmula 1 debería aportar respuestas a ciertas preguntas o hipótesis para poder evaluar y refinar las estrategias consideradas.

### **2.2.1. Hipótesis sobre los datos**

El estudio y análisis de los datos de la Fórmula 1 permite formular diversas hipótesis, las cuales se verifican o refutan mediante las diferentes visualizaciones. Las principales hipótesis que se plantean al tener diferentes datos de distintas carreras a lo largo de los años son:

- Diferentes circuitos influyen de manera distinta en el rendimiento de los vehículos.
- El rendimiento de los pilotos está influenciado por los tiempos por vuelta, los pit stops y la elección de compuestos de neumáticos.
- Los compuestos más blandos ofrecen mejores tiempos de vuelta.
- Las diferentes características de los equipos tienen un impacto significativo en los tiempos de vuelta.
- Las posiciones de salida influyen en la posición final y en los tiempos de vuelta.

Estas hipótesis orientan las tareas a realizar durante el análisis de los datos y las visualizaciones necesarias, lo que permite a los usuarios verificar posibles tendencias.

### 2.2.2. Requerimientos Funcionales

De las hipótesis enunciadas, a continuación se refinan unos ciertos requerimientos generales deseables en toda aplicación que trate con ellos. Son los siguientes:

- **REQ1:** Debe permitir la visualización de los datos de las competiciones de F1 recogidos durante años organizados según los años de competición de manera cronológica, mostrando fechas, ubicaciones y resultados clave, además de poder acceder a detalles de las carreras.
- **REQ2:** Debe proporcionar actualizaciones dinámicas, a tiempo real, según las consultas asegurando una experiencia fluida para el usuario.
- **REQ3:** Debe permitir la generación de gráficas interactivas según diferentes variables, como pilotos, carreras, competiciones y visualizar detalles concretos en diferentes gráficas como pitstops.
- **REQ4:** Debe incluir funciones de interacción directa en las gráficas o tablas, para ofrecer información contextual detallada (por ejemplo, al pasar el **mouse** sobre un elemento).
- **REQ5:** Debe garantizar la exportación de gráficas en formatos estándar para su uso externo.

### 2.2.3. Requerimientos No Funcionales

Además, se deben cumplir una serie de requisitos no funcionales básicos que se presentan a continuación:

- **REQ6:** Debe contar con un sistema de navegación intuitivo y bien estructurado que facilite al usuario moverse entre diferentes secciones de la aplicación, como análisis de datos, pilotos o constructores.
- **REQ7:** Las diferentes gráficas y funcionalidades deben ser responsive"que permita la compatibilidad con múltiples dispositivos y navegadores
- **REQ8:** Las diferentes funcionalidades deben funcionar de manera eficiente y rápida, consiguiendo una visualización más interactiva en tiempo real.
- **REQ9:** Las diferentes gráficas deben ser de fácil visualización para facilitar su utilidad para cualquier tipo de usuario.
- **REQ10:** Las diferentes gráficas deben ser extensibles dinámicamente, permitiendo al usuario realizar *zoom* o ajustar el nivel de detalle en caso de que no se visualicen correctamente.

Estos requerimientos están diseñados para proporcionar una herramienta o aplicación que no solo facilite la comprensión de los datos, sino que también enriquezca la interacción del usuario, combinando visualización, interactividad en el análisis.

## 2.3. Datos de la Fórmula 1

En este contexto, se generan una gran cantidad de datos a lo largo de cada carrera, desde tiempos de vuelta (TS) y pitstops (PS) hasta estrategias de neumáticos. En la actualidad, hay diversas API (Application Programming Interfaces) que permiten acceder a estos datos de manera estructurada, algunas son gratuitas y de código abierto, mientras que otras requieren una suscripción.

En resumen, en la Tabla 2.3, se destacan como las mejores opciones **OpenF1**, **FastF1** y **f1DataR** por su combinación de accesibilidad, precisión y funcionalidades avanzadas.

API	OpenSource	TS	PS	Fotografías	Información Detallada
OpenF1	✓	✓	✓	✗	✓
Ergast	✓	✓	✓	✗	✗
Fast-F1	✓	✓	✓	✗	✓
f1dataR	✓	✓	✓	✗	✓
SportMonks	✗	✗	✗	✗	✗

Tabla 2.3: Comparativa de APIs para datos de la Fórmula 1. Información Detallada incluye datos técnicos y específicos como sensores de velocidad en diferentes sectores, tipos de neumáticos utilizados, tiempos de vuelta, y otros aspectos clave

## 2.4. Chatbots para Visualización de Datos

En la actualidad, los chatbots no solo cumplen con funciones de interacción conversacional, sino que también pueden realizar análisis avanzados y generar visualizaciones basadas en datos. Herramientas como **OpenAI**, **NL4DV**, **ClaudeAI** y **Llama** destacan por su capacidad de combinar procesamiento de lenguaje natural con análisis de datos.

En la Tabla 2.4 se sintetizan las ventajas e inconvenientes principales de estos LLM. Se han evaluado diferentes aspectos de cada LLM, teniendo en cuenta los requisitos en la Sección 2.2.2, donde se describen las características específicas que son necesarias para nuestras funcionalidades. Primero, se ha analizado la capacidad de generar visualizaciones directamente (**RG**), una herramienta clave para proporcionar representaciones gráficas de los datos sin depender de ninguna herramienta externa. Además, se ha detallado la capacidad de generar respuestas en forma de Texto (**RT**). También se ha considerado la Respuesta Conversacional (**RC**), que evalúa la habilidad de cada LLM para interactuar con el usuario, brindando respuestas precisas y útiles. La documentación detallada (**DD**) es otro aspecto fundamental, ya que una documentación clara y completa, facilita la integración de los LLM en una aplicación externa. Y por último, se ha detallado si las herramientas son **Gratuitas**, y su capacidad de Manejar Múltiples Archivos, permitiendo procesar varias fuentes de datos al mismo tiempo.

Herramienta	RG	RT	RC	DD	Gratuita	MMA
OpenAI	✗	✓	✓	✓	✗	+1
OpenAI Assistant	✗	✓	✓	✓	✗	+1
NL4DV	✓	✗	✓	✓	✗	1
ClaudeAI	✗	✓	✓	✓	✗	+1
Llama	✗	✓	✓	✗	✓	+1

Tabla 2.4: Comparativa de herramientas para visualización y análisis de datos mediante chatbots. RG (Respues imágenes de gráficos), RT (Respuesta en forma de Texto), RC (Respuesta Conversacional), DD (Documentación Detallada), Gratuita (disponibilidad gratuita), MMA (Manejo de Múltiples Archivos: +1 = Varios Archivos, 1 = Único Archivo).

En resumen, la Tabla 2.4 presenta una comparación entre las distintas LLM. OpenAI y ClaudeAI se destacan por sus completas capacidades de respuesta conversacional y manejo de múltiples archivos, aunque ambos implican un coste. Por otro lado, en lo que respecta a la generación de gráficas a partir de consultas, NL4DV emerge como una buena alternativa, aunque con un coste menor y sin la capacidad de ofrecer respuestas textuales. En cambio, otras herramientas como Llama que son gratuitas, presentan limitaciones en cuanto a la documentación, ya que no hay suficiente información disponible para su implementación en proyectos donde texto y visualizaciones conviven conjuntamente.

### 2.4.1. Estado actual de las aplicaciones de la Fórmula 1

En el ecosistema actual, hay diferentes aplicaciones enfocadas en el análisis y la visualización de datos de la Fórmula 1. Estas herramientas están diseñadas para proporcionar a los usuarios expertos unos recursos avanzados que les ayude a comprender mejor el rendimiento de los pilotos, equipos y carreras. Sin embargo, cada aplicación tiene características particulares que las distinguen entre cada una de ellas.

- **F1 Live Timing [14]:** La plataforma oficial de la Fórmula 1 proporciona datos en tiempo real y cuenta con características gratuitas, como la comparación entre pilotos y la visualización de sus posiciones en el circuito actual. Sin embargo, funciones avanzadas como datos de sectores, segmentos, y neumáticos están disponibles únicamente para los usuarios suscritos.
- **F1 Tempo [15]:** Una plataforma enfocada en proporcionar datos completos sobre carreras y sesiones específicas a partir del 2018, incluyendo tiempos de vuelta y telemetría. Permite a los usuarios llevar a cabo un análisis detallado, mostrando gráficas comparativas y telemetría específica. Su diseño intuitivo y dinámico la hace una herramienta perfecta tanto para los analistas como para los usuarios expertos que desean realizar un seguimiento exhaustivo de las carreras.
- **Rapit [16]:** Página web dedicada a la visualización de datos mediante gráficos interactivos sobre carreras y sesiones específicas a partir del 2018. Presenta información de diferentes maneras como la tabla de clasificación, vueltas rápidas y un análisis visual detallado a través de gráficos que muestran los tiempos de vuelta, las estrategias de neumáticos y los sectores del circuito. Su enfoque en la representación visual de datos la convierte en una opción ideal para usuarios que buscan análisis claros y concisos.
- **PitWall [17]:** Una aplicación que centraliza información de la Fórmula 1 con datos disponibles desde su creación en el 1950. En su sección de análisis, los usuarios pueden comparar vueltas de una carrera con los datos a partir del 1996 o consultar informes detallados de cada carrera de un piloto. Este informe incluye la evolución de la posición en la carrera, los mejores tiempos de práctica y clasificación, los tiempos de carrera por vuelta y los pitstops. Aunque su interfaz no es la más atractiva visualmente, su valor radica en la amplitud y profundidad de los datos que ofrece, convirtiéndola en una herramienta interesante tanto para los usuarios expertos como para los aficionados casuales.
- **F1 Dash [18]:** Una plataforma para continuar las carreras de la Fórmula 1 en tiempo real. Ofrece una tabla detallada que incluye la posición de los pilotos, el tipo de neumáticos que utilizan, la activación de DRS, el gap con respecto al siguiente piloto, los tiempos de vuelta y de sector, así como las velocidades alcanzadas. También, se puede seguir la posición de cada piloto en su mapa interactivo del circuito. Además, puedes escuchar las radios entre ingenieros y pilotos, y ver la información sobre el estatus de la pista según la FIA. Esta página web está diseñada para ofrecer datos de manera gratuita y accesible, y puede ser una variable para poder continuar

las carreras, esta herramienta es ideal para los aficionados casuales sin acceso a la imagen de las carreras.

- **Formula1 Dashboard [19]:** Una herramienta diseñada para analizar datos exclusivamente de la temporada actual, ofreciendo estadísticas detalladas y visualizaciones interactivas. Incluye secciones diferentes como calendario, pilotos, consistencia, escuderías, pit stops, actualizaciones técnicas, elementos usados y costes por accidentes. Cada sección presenta un conjunto detallado de gráficos y visualizaciones interactivas que facilitan la comparación y en análisis del rendimiento a lo largo de la temporada. Esta plataforma es ideal para quienes buscan una visión global de la temporada, más que un enfoque en carreras individuales, convirtiéndola en una solución versátil para usuarios expertos y aficionados casuales.

En la Tabla 2.5, se sintetiza cómo cada aplicación existente cumple o no con los requerimientos definidos anteriormente. Aunque algunas aplicaciones actuales como **F1 Tempo** o **Rapit**, destacan en ciertos aspectos específicos, ninguna ofrece una solución integral que cubra todos los requerimientos funcionales. Esto refuerza la importancia de desarrollar una aplicación que no solo cumpla con los requerimientos establecidos, sino que también ofrezca una experiencia única al usuario.

Apps	REQ1	REQ2	REQ3	REQ4	REQ5	REQ6	REQ7	REQ8	REQ9	REQ10
F1 Live Timing	✗	✓	✓	✗	✗	✓	✓	✓	✓	✗
F1 Tempo	✗	✓	✓	✓	✗	✓	✓	✓	✓	✓
Rapit	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓
PitWall	✓	✗	✓	✗	✗	✓	✓	✗	✓	✗
F1 Dash	✗	✓	✗	✗	✗	✓	✓	✓	✓	✗
Formula1 Dashboard	✓	✓	✗	✓	✗	✓	✗	✗	✓	✗

Tabla 2.5: Comparativa de requerimientos con las aplicaciones existentes

Aunque estas aplicaciones proporcionan herramientas útiles, los requerimientos antes mencionados incorporan funciones exclusivas que fusionan visualización, interactividad, análisis y chatbot. Esto no solo permite competir con las aplicaciones actuales, sino que también brinda una experiencia más personalizada y ajustada a las necesidades de los usuarios.

En conclusión, F1 Tempo y Rapit, por sus capacidades de generación de gráficos, son las aplicaciones más destacadas. Sin embargo, ninguna de las aplicaciones analizadas en la Tabla 2.5 cuenta con la integración de un chatbot ya sea para consultar explicaciones sobre los datos, ya sea para interactuar con visualizaciones.

## 2.4.2. Retrieval Augmented Generation

Una reciente estrategia de diseño de chatbots con más habilidades es el sistema basado en **RAG** [20] es un enfoque avanzado en el procesamiento de lenguaje natural (NLP), creado para mejorar las habilidades de los modelos generativos al incorporar técnicas de recuperación de información. Este modelo ha sido desarrollado para abordar las limitaciones de los modelos, especialmente en lo que respecta a la gestión de información reciente o muy específica.

El RAG combina dos procesos claves para mejorar la precisión y relevancia de las respuestas que son producidas por los modelos, ver Figura 2.4.

- **Recuperación de información:** Utiliza diversas herramientas como motores de búsqueda, bases de datos o APIs para obtener documentos o datos relevantes que enriquezcan el contexto de una consulta del usuario.
- **Generación de la respuesta:** Combina el contexto con las capacidades generativas de nuestro modelo para producir una respuesta clara y detallada.

Este enfoque permite que se pueda ajustar a situaciones en las que es necesario incluir respuestas actualizadas, y evitar la dependencia total de su entrenamiento anterior.

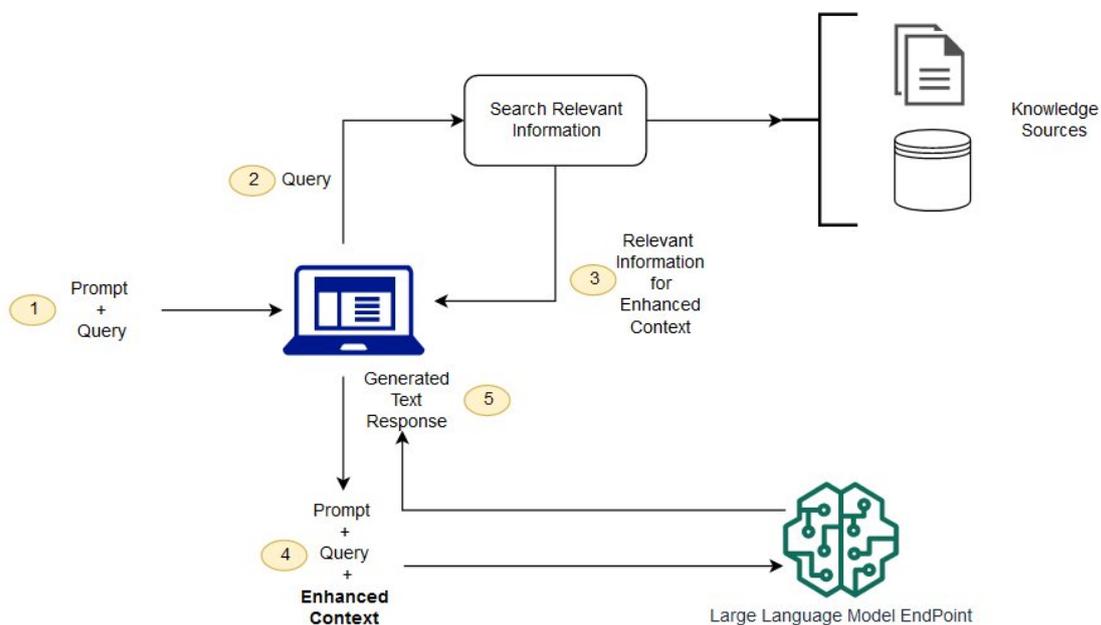


Figura 2.4: Flujo conceptual del RAG con el LLM. Foto extraída de Amazon.

En el contexto de la visualización de datos, el RAG se presenta como una técnica eficaz para abordar consultas complejas que requieren combinar datos dinámicos con capacidades explicativas avanzadas. Esto es especialmente valioso en aplicaciones donde se manejan grandes volúmenes de datos y necesitas proporcionar información actualizada.

Una de sus principales ventajas es el **acceso a información reciente**, lo que permite generar respuestas y visualizaciones basadas en datos actualizados, superando las limitaciones de los modelos ya entrenados que no cuentan con este conocimiento reciente. Además, el RAG **mejora la precisión** al integrarse con información externa al modelo generativo, y así poder garantizar una mayor relevancia y calidad en las respuestas. Por último, su **adaptabilidad** lo convierte en una herramienta versátil, ya que puede conectarse con diferentes herramientas y permitiendo ajustarse a las necesidades específicas de los usuarios.

En resumen, el RAG no únicamente mejora la calidad y relevancia de las respuestas en sistemas de visualización de datos, sino que también aporta flexibilidad en la creación, convirtiéndolo en una técnica esencial.

## 2.5. Conclusiones

En conclusión, una posible opción para implementar un sistema que cumpla los requerimientos descritos en la Sección 2.2.2, es emplear un enfoque mixto que combine un **Sistema basado en Reglas y Retrieval Augmented Generation**, visto en la Sección 2.4.2. Este enfoque híbrido permite estructurar las funcionalidades y visualizaciones de los datos de las APIs vistas en la Tabla 2.3 definiendo una serie de reglas que traduzcan las intenciones del usuario al tipo de petición concreto, garantizando una respuesta textual o visual según el caso. Por otro lado, utilizar RAG, que permite enriquecer el LLM con datos concretos, complementa el sistema al proporcionar flexibilidad y acceso a información de forma más dinámica y no tan tabulada.

Además, en la Tabla 2.4, se sintetizan varias herramientas potentes para la visualización y análisis de datos mediante queries. Entre ellas, **OpenAI** y **NL4DV** sobresalen por sus capacidades específicas. Para el presente TFG, se considera una combinación de ambas: **OpenAI** es la opción principal para la búsqueda de información, explicarla y ofrecer respuestas detalladas de manera conversacional, gracias a su avanzada integración y su capacidad para manejar múltiples archivos. Por otro lado, **NL4DV** se destaca como la herramienta ideal que se basa en un RAG de forma implícita, para la generación directa de gráficos, permitiendo crear imágenes de gráficos listas para usar. Esta combinación genera una solución integral que maximiza tanto la interacción con el usuario como la visualización de datos, y brindando una experiencia única al usuario.

# Capítulo 3

## Análisis

En este capítulo se lleva a cabo la fase de análisis del proyecto comenzando con la identificación de los distintos usuarios y los casos de uso. A continuación, se presenta el storyboard que incluye los diagramas visuales que representan las diversas funcionalidades. Finalmente, se incluye una sección dedicada al Chatbot y a los Datos, donde examinamos los requisitos específicos para su incorporación en el proyecto.

### 3.1. Casos de Uso

#### 3.1.1. Perfiles de Usuario

La aplicación a diseñar está orientada a dos tipos de usuarios, definidos según su nivel de experiencia y objetivos con la aplicación. Para entender mejor las necesidades y objetivos de los usuarios de la aplicación, se han definido los perfiles representativos basados en las posibles prácticas del diseño centrado en el usuario (DCU). Este enfoque permite personalizar las funcionalidades y mejorar la experiencia general de la aplicación [21].

- **Usuario Casual:** Este tipo de usuario tiene un interés general en la Fórmula 1, y busca una interfaz sencilla e intuitiva para explorar la información básica de las competiciones. Su interacción no requiere conocimientos previos, y únicamente desea aprender e informarse de los campeonatos de Fórmula 1, ver la Figura 3.1.



**Alba Fernandez**  
**32 Años, Valencia**  
**Sexo: Mujer**  
**Profesora de Secundaria**

**CONTEXTO:**  
 Alba es una profesora de secundaria, donde sus alumnos son unos aficionados a la Fórmula 1. Aunque ella no es una experta, le interesa seguir las carreras ocasionalmente para poder conectar con sus estudiantes. Le gusta aprender datos básicos de las carreras, como los ganadores de las carreras, los pilotos y escuderías actuales.

**OBJETIVOS:**  
 Alba busca información clara y bien estructurada sobre la Fórmula 1, incluyendo resultados de las carreras y algunas gráficas simples. Su intención es poder conversar sobre el tema de manera informada y disfrutar de las carreras de forma casual sin tener que hacer análisis complejos.

**FRUSTACIONES:**  
 A Alba no le gusta cuando las aplicaciones de Fórmula 1 son difíciles de usar o requieren mucho tiempo para aprender a navegar entre las diferentes pantallas. Además, le resulta frustrante que la mayoría de las herramientas estén diseñadas para usuarios más expertos, dejando de lado a quienes solo tienen un interés casual por el deporte.

**NECESIDADES:**  
 Alba necesita una aplicación que le permita explorar las estadísticas de las carreras de forma sencilla, con un diseño visualmente atractivo. También quiere acceder a una información básica sin perder tiempo configurando muchos filtros, y así disfrutar de la navegación intuitiva que no requiera conocimientos previos ni experiencia en análisis.

Figura 3.1: Ejemplo de Usuario Casual

- **Usuario Experto:** Este usuario está familiarizado con el análisis de datos de la Fórmula 1 y busca realizar comparaciones complejas y detalladas, utilizando herramientas avanzadas para obtener conclusiones a las distintas hipótesis que se plantea sobre los datos. El usuario podrá analizar los datos usando todas las funcionalidades de la aplicación, ver la Figura 3.2.



**Alejandro Martinez**  
**27 Años, Barcelona**  
**Sexo: Hombre**  
**Analista de Datos**

**CONTEXTO:**  
 Alejandro trabaja como analista de datos a tiempo completo en una empresa de tecnología. En su tiempo libre, sigue la Fórmula 1 desde su adolescencia. Su pasión por las carreras va más allá del entrenamiento, le encanta analizar el rendimiento de los pilotos y equipos, explorar estadísticas y entender las estrategias de cada carrera.

**OBJETIVOS:**  
 Alejandro quiere realizar análisis detallados sobre el rendimiento de pilotos y constructores en diferentes circuitos y temporadas mediante diferentes gráficas para así poder documentar sus conclusiones y compartirlas con sus amigos aficionados. También quiere ver como afectan algunos factores como los neumáticos y los pit stops a cada piloto.

**FRUSTACIONES:**  
 Alejandro le frustra tener que usar múltiples páginas web y herramientas para crear sus análisis, ya que cada una tiene interfaces diferentes y poco intuitivas, lo que le complica algunas tareas. También, la mayoría de aplicaciones no le permiten personalizar sus gráficas o pasan por alto detalles importantes. Además, odia los procesos de carga largos y tediosos, y si desea incluir algún gráfico en su documento, no puede descargarlos.

**NECESIDADES:**  
 Alejandro busca una solución que le permita acceder a toda la información sobre las carreras en una única plataforma, además de contar con herramientas avanzadas para crear gráficas y exportarlas en su documento. También desea que la interfaz sea intuitiva, facilitando tanto la exploración rápida como un análisis más detallado, y que sea compatible con diferentes navegadores.

Figura 3.2: Ejemplo de Usuario Experto

### 3.1.2. Casos de Uso

A partir de los requerimientos generales definidos en la sección 2.2 y de los perfiles de usuarios en la sección anterior, a continuación se describen los casos de uso concretos de la aplicación a desarrollar.

#### Usuario Casual

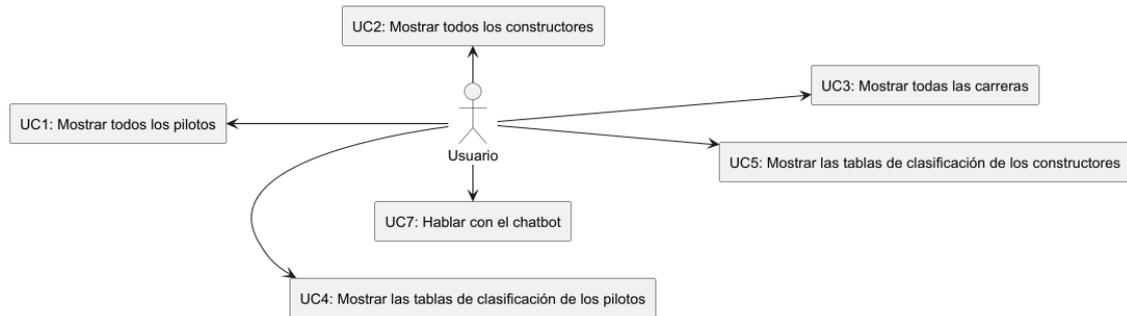


Figura 3.3: Casos de Uso para Usuario Casual.

Para el usuario Casual, se han identificado seis grandes grupos de casos de uso: **Mostrar todos los pilotos**, **Mostrar todos los constructores**, **Mostrar todas las carreras**, **Mostrar las tablas de clasificación de los pilotos**, **Mostrar las tablas de clasificación de los constructores** y **Hablar con el chatbot**.

A continuación, se detallan los casos de uso específicos dentro de cada grupo detallando los filtros y funcionalidades que se pueden aplicar en cada uno de ellos:

**UC1. Mostrar todos los pilotos:** El usuario tiene acceso a ver la información de cada driver (a partir de aquí en adelante usaremos driver para referenciar a un conductor) como nombre, país, abreviación.

- **UC1.1 Aplicar filtros de un año en específico:** El usuario ha de poder seleccionar la temporada específica para ver los drivers de esa temporada.

**UC2: Mostrar todos los constructores:** El usuario tiene acceso a ver la información de cada constructor como el nombre, el país, el color y su bandera.

- **UC2.1 Aplicar filtros de un año en específico:** El usuario ha de poder seleccionar la temporada específica para ver los constructores de esa temporada.

**UC3: Mostrar todas las carreras:** El usuario tiene acceso a ver todas las races (a partir de aquí en adelante, usaremos races o race para referenciar a una carrera) ordenadas cronológicamente.

- **UC3.1 Aplicar filtros de un año en específico:** El usuario ha de poder seleccionar la temporada específica para ver las carreras de esa temporada.

- **UC3.2 Abrir la información de una race:** El usuario puede abrir la información detallada de una carrera específica para realizar un análisis sobre esa carrera.
  - **UC3.2.1 Visualizar la gráfica comparativa de vueltas por constructores:** El usuario puede visualizar la gráfica comparativa de las vueltas realizadas por los diferentes constructores durante esa carrera.
  - **UC3.2.2 Visualizar la gráfica por posiciones:** El usuario ha de poder visualizar una gráfica de como cambiaron las posiciones de los pilotos seleccionados durante la carrera.
    - **UC3.2.2.1 Mostrar pitstops:** El usuario ha de poder seleccionar si quiere mostrar los pitstops del piloto seleccionado dentro de la gráfica por posiciones.
  - **UC3.2.3 Visualizar la gráfica de tiempos y compuestos por piloto:** El usuario ha de poder generar una gráfica que muestre los compuestos de neumático y los tiempos por vuelta utilizados de un piloto seleccionado.
    - **UC3.2.3.1 Mostrar pitstops:** El usuario ha de poder seleccionar si quiere mostrar los pitstops del piloto seleccionado dentro de la gráfica de tiempos y compuestos.
  - **UC3.2.4 Visualizar la gráfica de tiempos por piloto:** El usuario ha de poder generar una gráfica que muestre los tiempos por vuelta utilizados de los pilotos seleccionados.
    - **UC3.2.4.1 Mostrar pitstops:** El usuario ha de poder seleccionar si quiere mostrar los pitstops de un piloto seleccionado dentro de la gráfica de tiempos.
  - **UC3.2.5 Visualizar la gráfica de distribución de las vueltas rápidas:** El usuario ha de poder visualizar una gráfica de distribución de tiempos de vueltas rápidas por cada piloto seleccionado y que muestre información clave como tiempos mínimos y máximos, mediana, cuartiles y densidad. También, debe permitir identificar a cada piloto seleccionado mediante la abreviación y colores en el gráfico.

**UC4: Mostrar las tablas de clasificación de los pilotos:** El usuario tiene acceso a las standings (a partir de aquí en adelante, usaremos standings para referenciar a las tablas de clasificación) de cada driver, organizado por cada gran premio.

- **UC4.1 Aplicar filtros de un año en específico:** El usuario ha de poder seleccionar la temporada específica para ver las clasificaciones de los pilotos de esa temporada.

**UC5: Mostrar las tablas de clasificación de los constructores:** El usuario tiene acceso a las standings de cada constructor, organizado por cada gran premio.

- **UC5.1 Aplicar filtros de un año en específico:** El usuario ha de poder seleccionar la temporada específica para ver las clasificaciones de los constructores de esa temporada.

**UC7: Hablar con el chatbot:** El usuario tiene acceso al chatbot para poder realizar consultas o generar gráficas.

- **UC7.1 Solicitar gráfica:** El usuario puede pedir al chatbot que genere una gráfica específica que no se encuentra en la aplicación
  - **UC7.1.1 Expandir la gráfica:** El usuario puede ampliar la gráfica generada por el chatbot para tener una mejor visión.
  - **UC7.1.2 Redirección en la página:** El usuario puede ser redirigido a alguna sección de la página en caso de que se encuentre una parecida a la que se ha solicitado.
- **UC7.2 Solicitar información:** El usuario puede realizar una consulta para solicitar información. El chatbot proveerá información sobre su consulta, y generará una respuesta basada sobre los datos encontrados.
- **UC7.3 Solicitar más detalles de ciertas partes de una gráfica:** El usuario puede solicitar información sobre una zona de una gráfica. El chatbot deberá detallar una explicación más detallada sobre la información solicitada.

## Usuario Experto

El usuario experto tiene los mismos casos de uso que el usuario casual. La única diferencia que hay entre estos dos usuarios es que el usuario experto podrá hacer el análisis de datos donde se añade el grupo correcto, como se observa en la Figura 3.4.

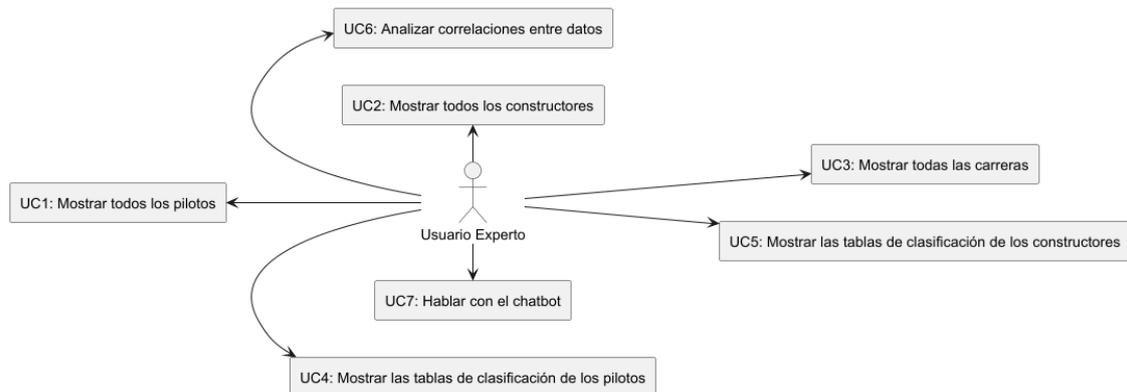


Figura 3.4: Casos de Uso para Usuario Experto.

**UC6: Analizar correlaciones entre datos:** El usuario tiene acceso a ver todas las races para poder realizar la gráfica de correlación.

- **UC6.1 Aplicar filtros de un año en específico:** El usuario puede seleccionar una temporada para analizar datos específicos de esa temporada.

- **UC6.2 Visualizar la correlación según las carreras seleccionadas:** El usuario puede generar una gráfica de correlación entre diferentes variables para analizar tendencias o patrones en los datos de las carreras seleccionadas.
- **UC6.3 Seleccionar todas las races:** El usuario puede seleccionar todas las carreras de forma rápida, sin tener que seleccionar una a una.
- **UC6.4 Deseleccionar todas las races:** El usuario puede deseleccionar todas las carreras de forma rápida, sin tener que seleccionar una a una.

## 3.2. Diseño gráfico y story board

Una vez introducidos los casos de uso a continuación se detalla su diseño gráfico de la aplicación así como la navegación entre ellos a través de su story board. En la Figura 3.5 se muestra la interfaz principal de la aplicación, donde se destacan tres elementos clave:

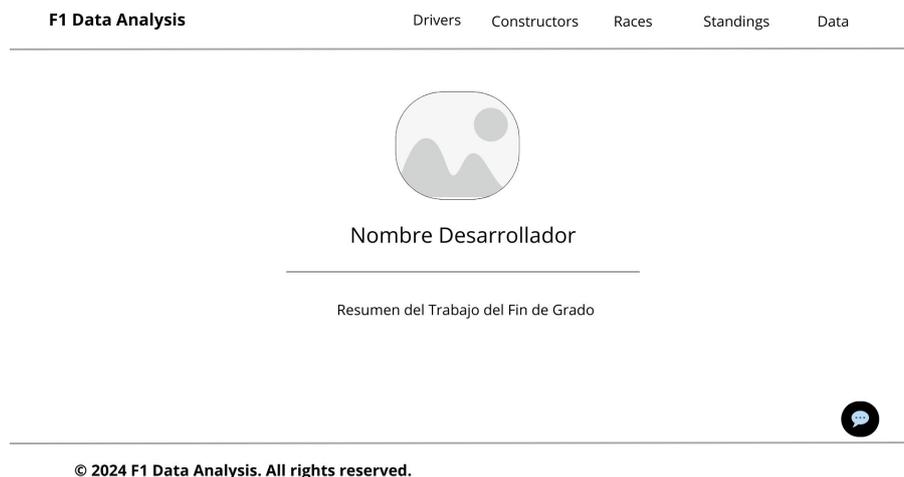


Figura 3.5: Sketch de la página principal de la aplicación.

1. **Secciones principales:** Corresponden a las diferentes áreas de los casos de uso descritos en la Sección 3.1.2.
2. **Botón del chatbot:** Situado en la parte inferior derecha, este botón permite al usuario interactuar con el chatbot, relacionado con el caso de uso UC7 Hablar con el chatbot.
3. **Main Page:** En el centro de la página principal se incluye el resumen, acompañada de una foto del desarrollador.

### 3.2.1. UC1. Mostrar todos los pilotos

El diseño de la Figura 3.6 muestra la interfaz propuesta para visualizar a todos los drivers, que incluye:

- Un filtro por año, permitiendo al usuario seleccionar la temporada específica que desea.
- Una tarjeta para cada piloto, que presenta información clave como su fotografía, nombre, país de origen, bandera y abreviación.
- Una disposición clara y ordenada que garantiza una experiencia intuitiva para todo tipo de usuario, ya sea casual o experto.

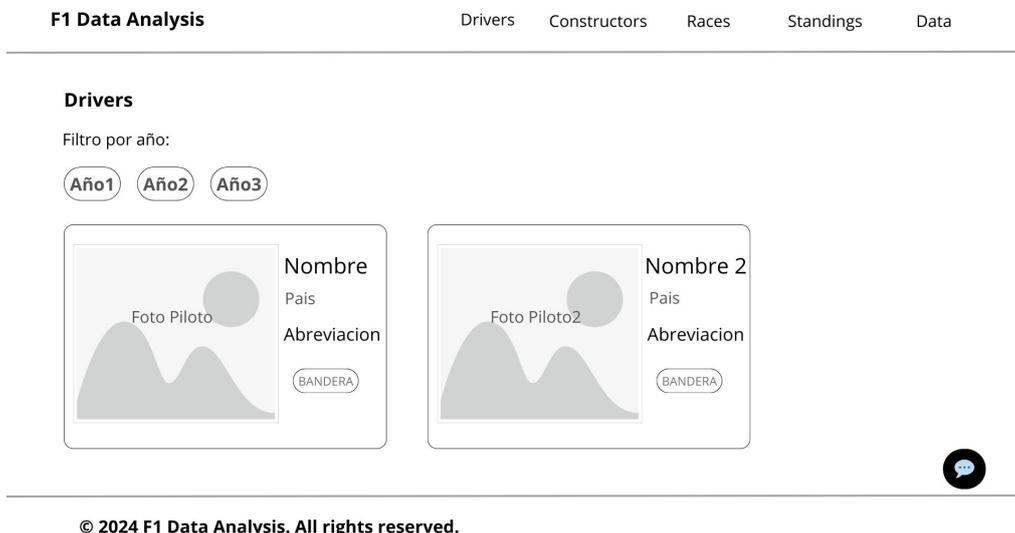


Figura 3.6: Sketch de todos los Drivers

### 3.2.2. UC2: Mostrar todos los constructores

El diseño de la Figura 3.7 presenta la interfaz para la visualización de todas las escuderías. En este sketch incluye:

- Un filtro por año, que permite al usuario seleccionar la temporada específica para explorar todos los constructores.
- Una tarjeta para cada constructor que contiene información clave, como su logo, nombre, color, país de origen y la bandera correspondiente.
- Una organización definida y ordenada, y creada para asegurar una experiencia fluida e intuitiva para todos los usuarios.

**Constructors**

Filtro por año:

Año1 Año2 Año3



© 2024 F1 Data Analysis. All rights reserved.

Figura 3.7: Sketch de todos los Constructores

### 3.2.3. UC3: Mostrar todas las carreras

Se muestra la interfaz propuesta para visualizar todas las races, ver la Figura 3.8:

- Un filtro por año, permitiendo al usuario seleccionar la temporada específica para explorar las diferentes carreras.
- Una tarjeta para cada carrera, que presenta información relevante como el nombre del gran premio, el circuito, la fecha de la carrera, la bandera y el país, y el mapa del circuito.
- Un botón de **información** que permite al usuario abrir la vista detallada de la carrera seleccionada, ver Figura 3.9.

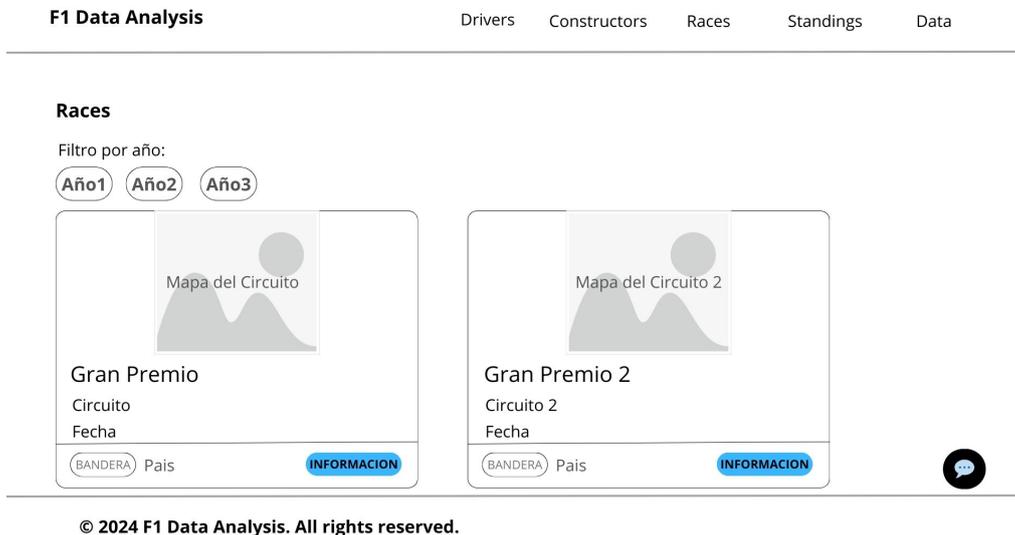


Figura 3.8: Sketch de todas las Races

Al seleccionar una carrera, se abre información sobre ella (Caso de Uso UC3.2) y contiene los siguientes detalles, mostrados en la Figura 3.9:

- El nombre del gran premio y el año en el que se celebra.
- Una tarjeta para cada piloto con su nombre, imagen y su posición final, ordenados correctamente.
- Botones para generar diferentes gráficas, la gráfica que quiera explorar el usuario: Gráfica de Vueltas por Constructores, Gráfica por Posiciones, Gráfica de tiempos y compuestos, Gráfica de tiempos y Gráfica de Distribución de las vueltas rápidas. Para poder visualizar estas gráficas se necesita tener un piloto o varios seleccionados.
- Botones para alternar entre los diferentes pilotos.

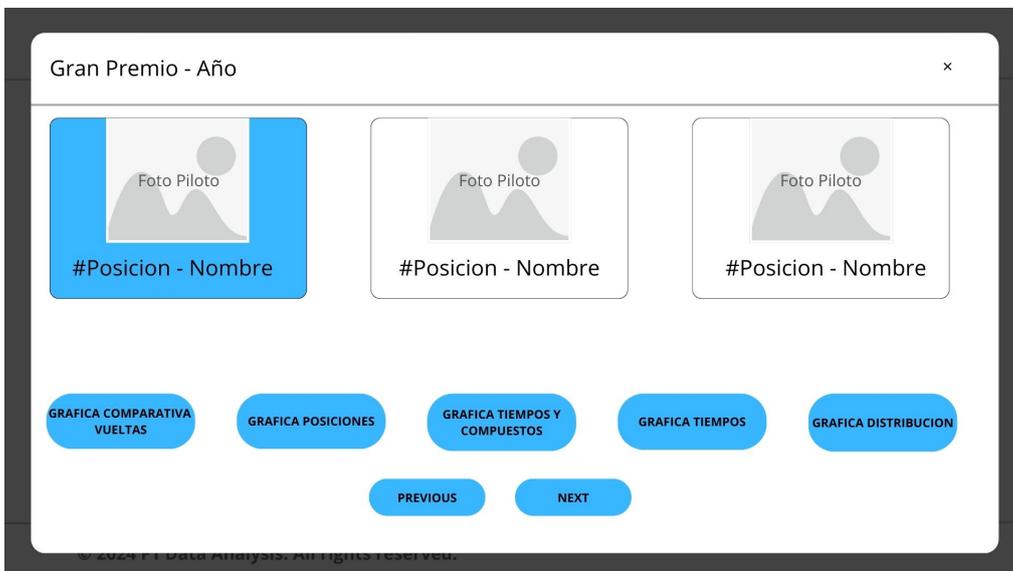


Figura 3.9: Sketch de los detalles de la Races.

Una vez seleccionada una de las gráficas, se abre su visualización interactiva (UC3.2.2, UC3.2.3, UC3.2.4 Visualizar la gráfica) de forma que el usuario puede solicitar mostrar los pit stops o no interaccionando en la misma gráfica, ver la Figura 3.10.



Figura 3.10: Sketch de la gráfica generada

### 3.2.4. UC4: Mostrar las tablas de clasificación de los pilotos

La Figura 3.11 presenta la interfaz para la visualización de las clasificaciones de los pilotos según los distintos años:

- Un filtro por año, que permite al usuario seleccionar la temporada específica para explorar las diferentes clasificaciones en la temporada.
- Una tabla donde hay los diferentes pilotos de la temporada, y los puntos que han conseguido en cada Gran Premio, y ordenado por clasificación.



Figura 3.11: Sketch de los Standings de los Drivers

### 3.2.5. UC5: Mostrar las tablas de clasificación de los constructores

De la misma manera que se mostraban las clasificaciones por pilotos, en la Figura 3.12 se muestra como se presenta la interfaz para las clasificaciones de los constructores:

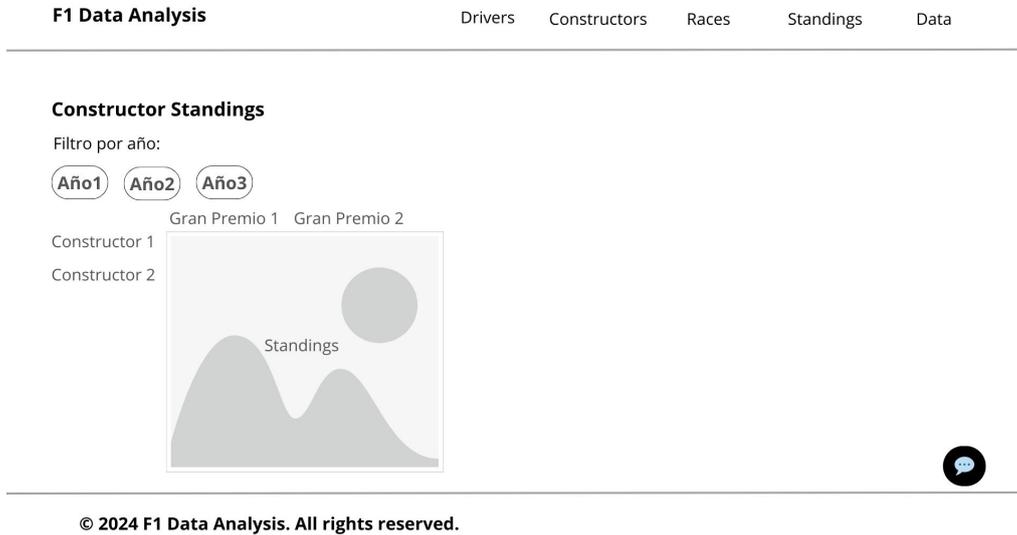


Figura 3.12: Sketch de los Standings de los Constructores

### 3.2.6. UC6: Analizar correlaciones entre datos

La propuesta para la selección de todas las carreras y poder generar el gráfico de correlación, mostrada en la Figura 3.13, incluye:

- Un filtro por año, que permite al usuario seleccionar la temporada específica para explorar las diferentes carreras de la temporada.
- Una tarjeta para cada carrera donde indica el Gran Premio, el mapa del circuito y la fecha de la carrera.
- Un botón para generar la gráfica de correlación tal y como se ha descrito en el caso de uso, y que muestre la siguiente imagen 3.14.
- Dos botones para ayudar al usuario a deseleccionar o seleccionar todas las carreras de forma rápida e intuitiva.

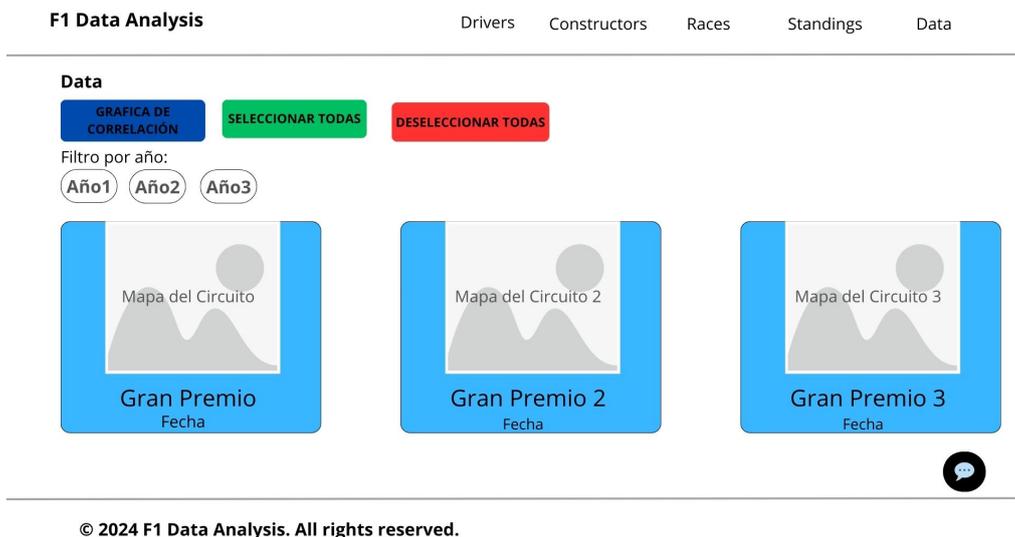


Figura 3.13: Sketch de la sección de Data Analysis



Figura 3.14: Sketch de la gráfica de correlación generada

### 3.2.7. UC7: Hablar con el chatbot

El diseño del chatbot, ver la Figura 3.15, busca ofrecer una interfaz interactiva y clara para poder reflejar un flujo de conversación entre el chatbot y el usuario.

- El chatbot da la bienvenida con un mensaje inicial.
- El usuario interactúa con el chatbot mediante un mensaje de texto donde puede escribir su consulta, y un botón de envío del mensaje.
- El usuario puede cerrar en cualquier momento la interacción con el chatbot mediante un botón.

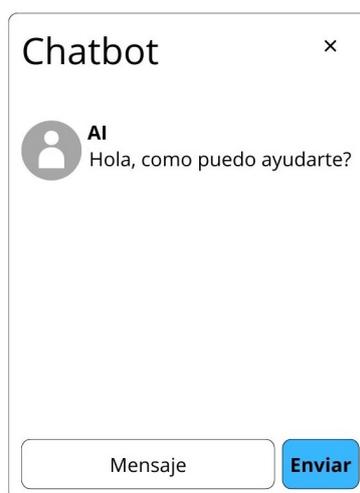


Figura 3.15: Sketch del Chatbot.

Cuando el usuario envía un mensaje al chatbot, este puede responder de distintas maneras según la consulta realizada:

- Solicitar gráficas no incluidas al inicio de la aplicación, caso de uso UC7.1. Por defecto, estas gráficas, ver la Figura 3.16 se generan en el interior de la caja del diálogo del chatbot. El usuario puede ampliar la gráfica (UC7.1.1).



Figura 3.16: Sketch de la Generación de Gráficas.

- Redirección en caso de que la gráfica solicitada, sea parecida a una que se encuentra en la aplicación (caso de uso UC7.1.2). En este caso, el chatbot requiere que el usuario confirme si quiere ver la página de la aplicación, ver diálogo de la Figura 3.17.

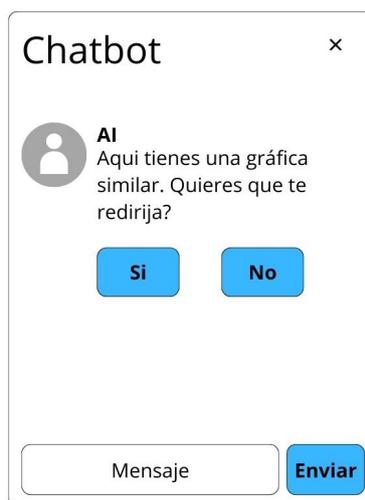


Figura 3.17: Sketch de la Redirección.

- El chatbot responde a una solicitud de información más compleja dando explicaciones textuales resumidas extraídas de fuentes externas (UC7.2).
- Pedir detalles específicos de las gráficas en ciertas posiciones interiores de una gráfica. En este caso, la interacción es mixta, ya que el usuario ha de apuntar a una cierta zona de la gráfica a la vez que pide información mediante texto al chatbot (UC7.3)

### 3.3. Propiedades del Chatbot

Una vez descrita la interacción del chatbot, a continuación se detallan ciertas propiedades específicas que tendrán su impacto en la aplicación. El objetivo del chatbot a diseñar es enriquecer la experiencia del usuario tanto experto como casual, proporcionando no solo una interacción conversacional, sino también la generación de gráficos personalizados y respuestas basadas en los datos.

#### 3.3.1. Propiedades del Chatbot

El chatbot debe cumplir con ciertas propiedades:

- **P1:** Debe ser capaz de generar tanto texto como nuevas gráficas.
- **P2:** Debe poder interpretar consultas según lo que se ha solicitado, incluyendo la información del ratón.
- **P3:** Debe recordar consultas previas.
- **P4:** Debe ser reactivo en las conversaciones, es decir, únicamente responder a las consultas de usuario sin proponer nuevos análisis.
- **P5:** Debe mantener información de diferentes fuentes como CSV, gráficas, descripciones resumidas en breves párrafos de internet.
- **P6:** Debe poder recomendar gráficas que se encuentran en la aplicación.

### **3.3.2. Impacto en la Aplicación**

Gracias a esta funcionalidad, la experiencia del usuario se ve mejorada al:

- Simplificar la interacción con los datos y las visualizaciones, ya que se evita la necesidad de navegar manualmente.
- Facilitar análisis avanzados para los usuarios expertos mediante la creación de nuevas gráficas.
- Ofrecer una nueva experiencia intuitiva para los usuarios casuales que solo desean obtener información rápida y precisa.

Estos beneficios no solo enriquecen la experiencia de interacción con el chatbot, sino que también genera una nueva herramienta para la visualización de datos, garantizando un acceso más eficiente a la información.

### **3.4. Datos**

En esta aplicación se deben analizar diferentes variables mostradas en distintos años, grandes premios, e implican diferentes pilotos, escuderías, valores de velocidad, neumáticos, entre otros valores. Los datos a procesar y analizar por la aplicación deben tener unas ciertas propiedades:

- Los datos han de ser accesibles de forma fácil y gratuita.
- Los datos no son dinámicos, es decir, son datos obtenidos a partir del 2018, que estarán residentes en la aplicación desde el inicio.
- Los datos han de incluir no solo valores de variables de las carreras como los nombres de los pilotos, tiempos, número de vueltas, etc. Si no que además han de incluir aspectos gráficos como banderas, colores en la escudería, fotos de pilotos y escuderías, etc.
- Los datos no debe contener valores nulos.
- Los datos han de estar uniformizados en cuanto al mismo número de variables en los distintos años así como en indexación a diferentes valores.

Estas propiedades van a guiar la elección de la fuente de datos y su pre-procesado para poder ser tratados por la aplicación.

# Capítulo 4

## Diseño

En este capítulo se presenta un análisis del diseño y el desarrollo de la aplicación. Empieza con una descripción de la arquitectura general del sistema. A continuación, se explica el diseño del chatbot y la estructura de la base de datos utilizada. Seguidamente, se detalla la interacción entre la visualización y la lógica. Finalmente, se describe el desarrollo de la aplicación, incluyendo la arquitectura y las características del backend.

### 4.1. Arquitectura general de la aplicación

La aplicación web que se ha desarrollado haciendo uso del patrón **Modelo-Vista-VistaModelo** (MVVM) [22], una versión adaptada a aplicaciones web del patrón Modelo-Vista-Controlador. La principal diferencia entre estas es que el patrón MVVM se centra en mejorar la interacción entre la vista y los datos a través de una sincronización automática entre la interfaz y la lógica, conocida como *data binding*.

El patrón MVVM se compone de tres componentes, ver la Figura 4.1:

- **Vista (View):** Encargada de establecer la estructura, el diseño y de la interfaz del usuario.
- **VistaModelo (ViewModel):** Es el enlace entre la vista y el modelo, responsable de manejar la lógica de la vista y facilitar el flujo de los datos.
- **Modelo (Model):** Contiene la lógica y los datos de la aplicación.

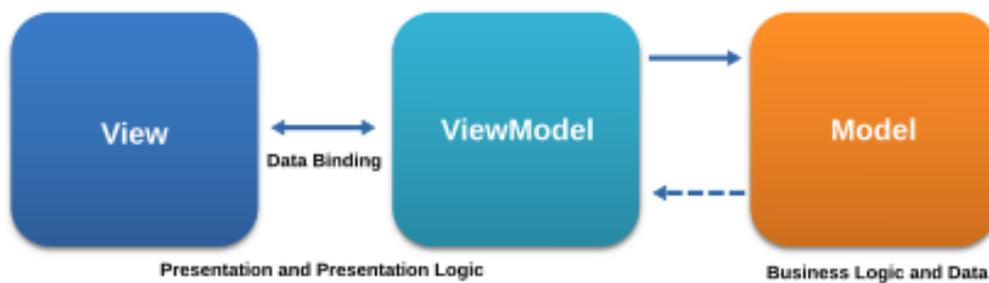


Figura 4.1: Patrón Arquitectónico del Modelo-Vista-VistaModelo. Foto extraída de Wikipedia.

En la Figura 4.2 se detalla la arquitectura general de la aplicación. Donde hay tres componentes, siguiendo el patrón anteriormente comentado en la Figura 4.1. Primero de todo, se encuentra la interacción entre los usuarios y el **frontend** que se encarga de gestionar la interfaz. Este componente se encarga de enviar solicitudes al **backend**, encargada de procesar las solicitudes y realizar consultas a la **base de datos**.

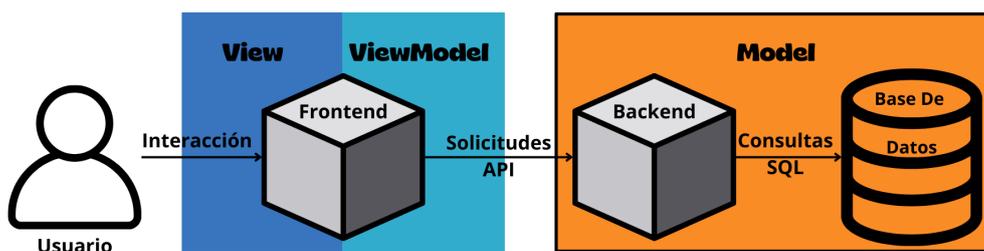


Figura 4.2: Arquitectura general de la aplicación

## 4.2. Modelo

La capa **Modelo** de la aplicación, comentada en la Sección 4.1. Este componente se encarga de procesar las solicitudes del Frontend y gestionar las llamadas a la Base de Datos. Este componente actúa como el núcleo que conecta y soporta el flujo de datos entre el **ViewModel** y la **View**. Además, será la responsable de la creación y funcionamiento del chatbot.

### 4.2.1. Diseño del Chatbot

El chatbot de la aplicación está diseñado para interpretar las diferentes entradas del usuario y generar respuestas basadas en el contexto y la solicitud exacta. Las diferentes entradas que utiliza el chatbot incluyen:

- **Texto del Usuario:** Incluye la consulta o query proporcionada por el usuario.

- **Contexto de la Visualización:** Incluye información sobre la gráfica actual registrada con la posición del ratón.
- **Historial de Consultas:** Permite al chatbot recordar todas las consultas anteriores realizadas por el usuario, para mantener un flujo de conversación coherente, y una creación de gráficas más personalizadas.

En cambio, el chatbot genera diferentes tipos de respuesta (ver Figura 4.3) que depende en función a la entrada procesada.

- **Generación de Gráficas:** Crea las diferentes visualizaciones basadas en consultas específicas del usuario.
- **Explicaciones Interactivas:** Proporciona detalles sobre las gráficas mostradas permitiendo al usuario profundizar en el análisis.
- **Redirección a Gráficas Existentes:** Sugerencias para explorar otras gráficas disponibles que podrían satisfacer las necesidades del usuario.
- **Búsqueda de Información Externa:** Integra información adicional obtenida mediante motores de búsqueda para ofrecer respuestas completas.

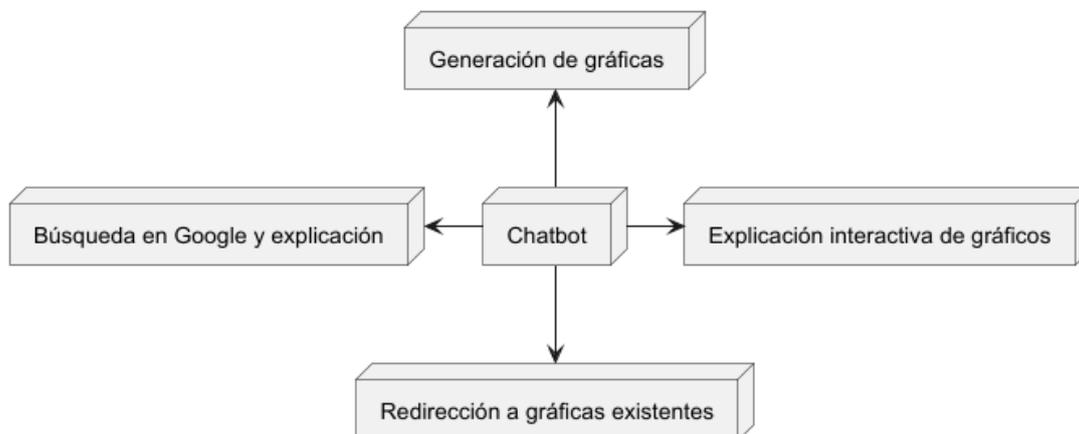


Figura 4.3: Diseño del chatbot: salidas principales.

Este chatbot para procesar las salidas, depende de un algoritmo para tomar dichas decisiones (sistema de reglas), y seguir un flujo lógico. Primero comienza con un análisis semántico de la consulta del usuario para identificar palabras clave que indiquen que acción tomar como generar una gráfica, explicar una visualización o buscar información externa. Cuando se haya identificado, la regla relativa a la opción identificada genera la salida dependiendo de esta acción, e indica que tipo de acción se ha escogido y así el Frontend poder gestionar dicha salida.

## 4.2.2. Diseño de la Base De Datos

La base de datos ha sido estructurada para contribuir todas las funciones principales. El modelo Entidad-Relación (E-R), ver en la Figura 4.4, representa las relaciones y los atributos de las entidades que forman la base de datos.

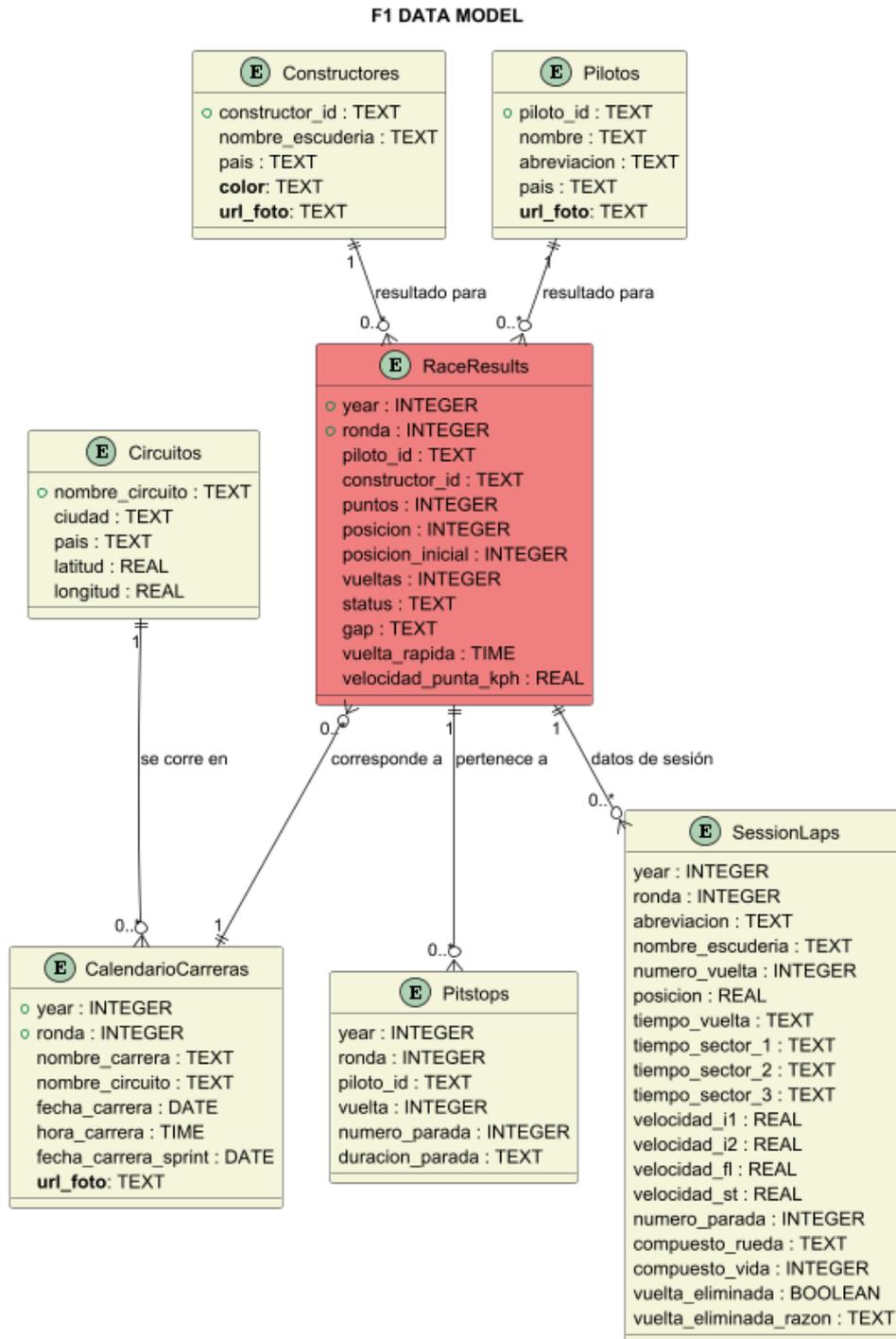


Figura 4.4: Diagrama E-R de la Base de Datos. Los tipos de relación han sido extraídas de PlantUML.

Las principales entidades de la base de datos son:

- **Pitstops:** Contiene información detallada sobre las paradas en boxes de los pilotos durante las carreras.
- **SessionLaps:** La tabla más importante, ya que con ella, podremos realizar todos los análisis de datos, gráficas. En ella, se registra todos datos de cada vuelta para cada uno de los pilotos en las sesiones de las carreras.
- **RaceResults:** Proporciona los resultados generales de las carreras.
- **Constructores:** Almacena información sobre los equipos.
- **Pilotos:** Contiene la información básica de los pilotos.
- **CalendarioCarreras:** Registra los detalles de cada carrera como el nombre de la carrera, la fecha y hora.
- **Circuitos:** Incluye información de todos los circuitos.

## 4.3. View y ViewModel

El diseño del **frontend** de la aplicación está basado en componentes lo que permite una estructura clara y organizada, cubriendo las dos capas **View** y **ViewModel** del patrón MMVV. Cada componente cumple una función específica dentro del proyecto, lo que facilita la escalabilidad y el mantenimiento del código.

### 4.3.1. View

La capa **View** es la encargada de la parte visual de la aplicación, y está diseñada para cumplir los requerimientos no funcionales vistos en la Sección 2.2.3. Para ello, se han tenido en cuenta los siguientes puntos:

- **Diseño de componentes:** Cada página es organizada en diferentes componentes individuales con diferentes funciones de la interfaz del usuario como menús, gráficas, tablas o filtros. Además de sintetizar el proceso del desarrollo, también proporciona un diseño más coherente e intuitivo para el usuario.
- **Renderización dinámica de los datos:** Una vez procesados, se integran en el componente para ser visualizados de manera dinámica, asegurando que el usuario tenga una interfaz interactiva sin necesidad de recargar la página.
- **Renderización del chatbot:** El chatbot se incorpora de forma interactiva en la capa de vista, brindando al usuario una experiencia intuitiva según la salida que se haya generado.

Analizando todas las gráficas necesarias descritas en la Sección 3.1.2, a continuación se detallan las decisiones escogidas para definir las diferentes gráficas que ayudaran a analizar las variables, según sus distintos tipos e información a extraer sobre ellas:

- **Gráficas para variables numéricas discretas:** Estas gráficas se centran en presentar datos con valores enteros, organizados en secuencias. Por ejemplo, la variable discreta "posición en carrera" es una candidata a visualizar con este tipo de gráfica. En la Figura 4.5 se muestran los diferentes cambios de posición de los pilotos a lo largo de la carrera. Cada piloto tiene el color asociado de su escudería.

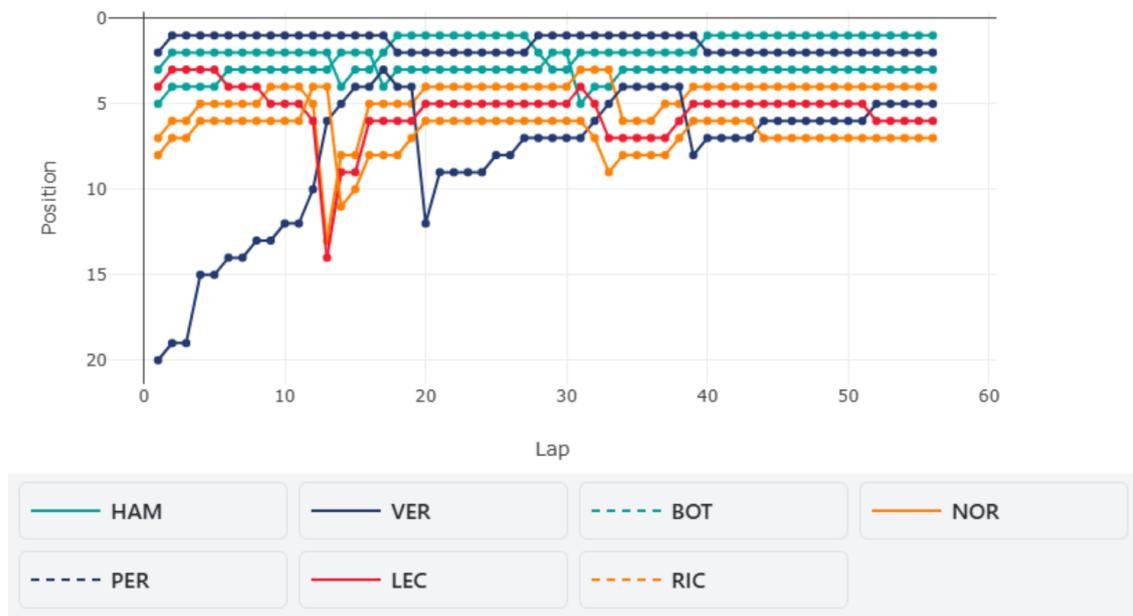


Figura 4.5: Ejemplo de Gráfica para variables numéricas discretas

- **Gráficas para promedios y variancias en variables continuas con muchas muestras:** Diseñadas para presentar información organizada en categorías específicas donde se examinan los promedios y las dispersiones entre las categorías. Un ejemplo donde se muestran los promedios de tiempos por vuelta de las diferentes escuderías en el año, como se puede ver en la Figura 4.6. Se ha escogido este tipo de gráfica cuando se realizan las medias de muchos datos de la variable continua. En este caso se están mostrando los promedios de los tiempos por vuelta según las escuderías, que implican los tiempos de todas las vueltas para todos los pilotos de la escudería.

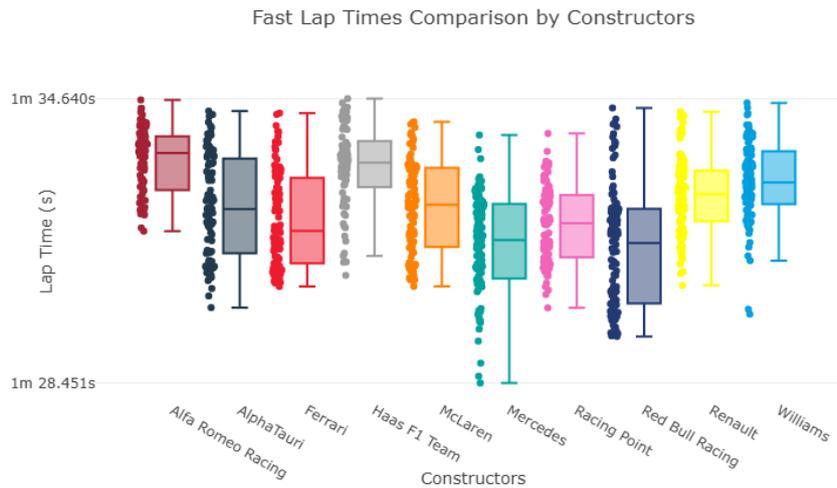


Figura 4.6: Ejemplo de Gráfica para datos de promedios y variancias en variables continuas.

- Gráficas para promedios y variancias en variables continuas con pocas muestras:** Estas gráficas, como la gráfica de la Figura 4.7, representan también datos agregados de medias y varianzas, donde se destaca la dispersión y los valores representativos de las categorías analizadas. En este caso se están mostrando los tiempos por vuelta de una carrera concreta para un conjunto de pilotos (en el eje X)

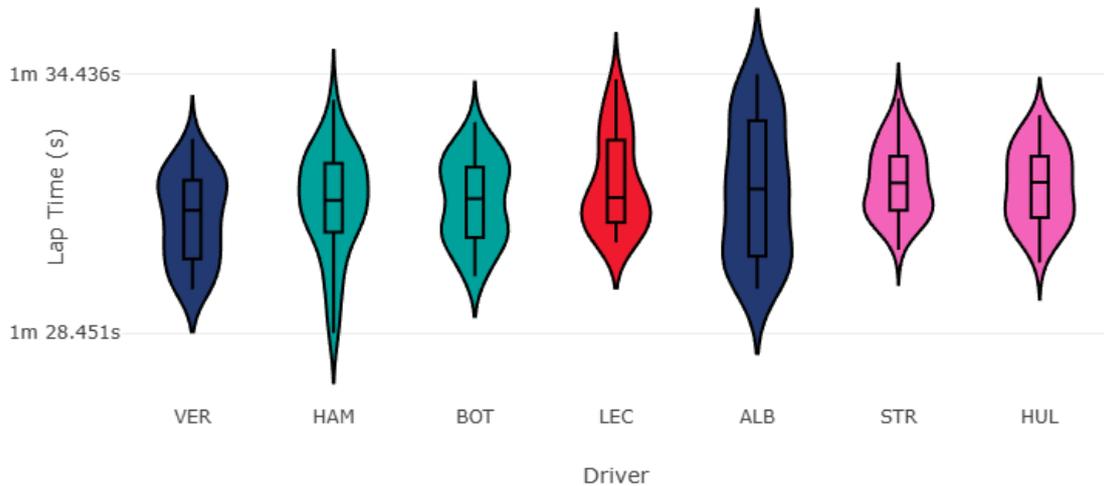


Figura 4.7: Ejemplo de Gráfica para variables de medias y varianzas

- Gráficas para variables continuas segregadas por valores nominales:** Representan categorías sin ordenar con gráficos de puntos sin secuencia temporal o jerarquía como la gráfica de la Figura 4.8. En esta gráfica se muestran para cada vuelta, el tiempo de vuelta coloreado por el tipo de neumático.

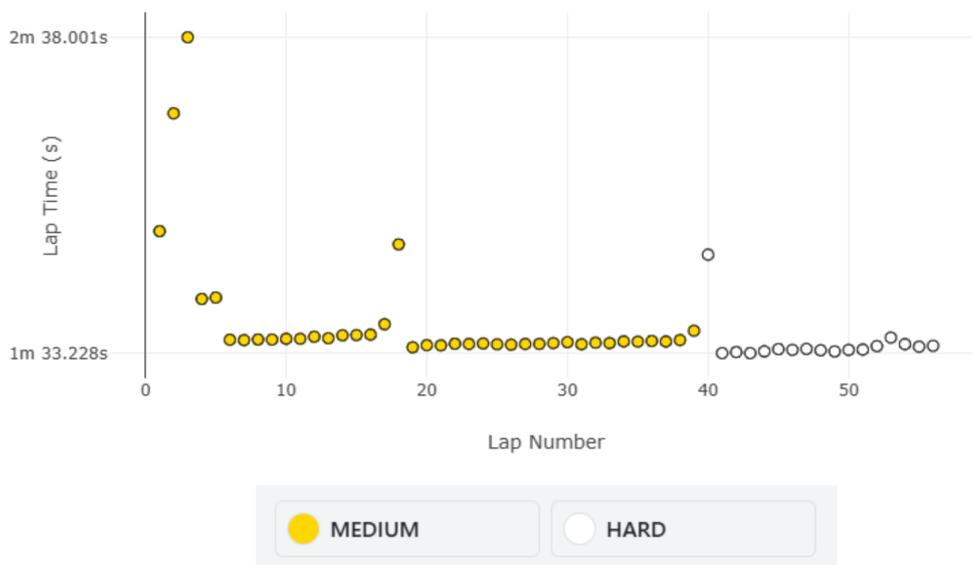


Figura 4.8: Ejemplo de Gráfica para variables continuas segregadas por valores nominales.

- Correlaciones:** Las matrices de correlación como por ejemplo para la gráfica de la Figura 4.9, permite analizar la relación entre las diferentes variables en un mapa de calor. Un azul más intenso implica una correlación más alta positiva y un valor blanco implica una ausencia de correlación. Esta gráfica permite detectar las variables que tienen más relación en una visualización global de los datos.

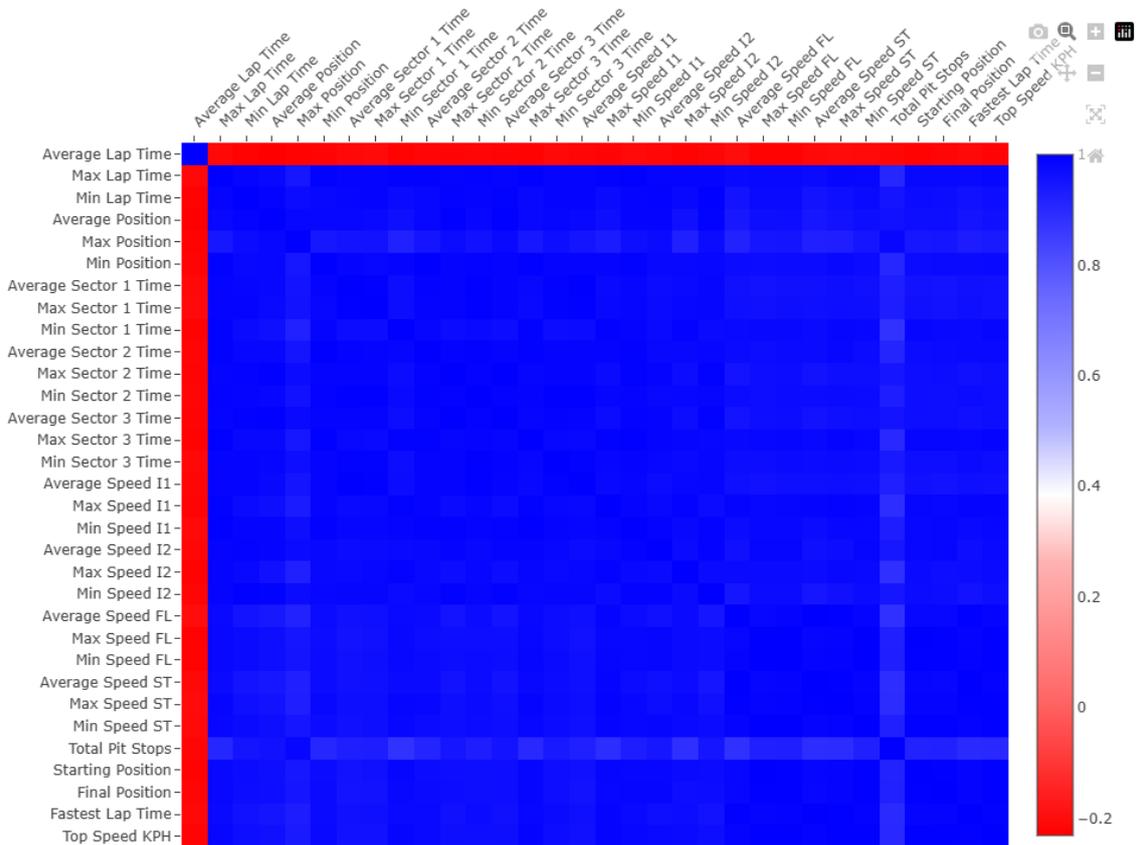


Figura 4.9: Ejemplo de Gráfica de Correlación.

- **Gráficas simples para generar histogramas, plots, etc.:** son gráficas que ofrecen una representación sencilla de la distribución de frecuencias de variables individuales, proporcionando una visión general rápida y efectiva de los datos. En general, son las gráficas que va a generar el chatbot (ver Figura 4.10).

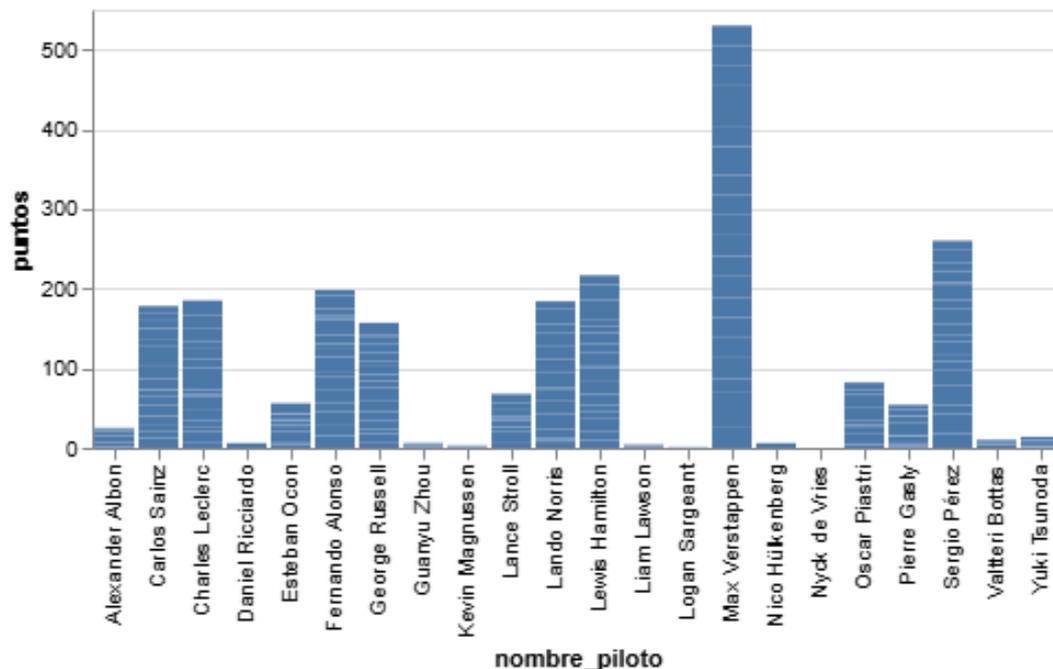


Figura 4.10: Ejemplo de Gráfica generada por el Chatbot.

### 4.3.2. ViewModel

En la parte del ViewModel, que también está incluida en el frontend, se reservan las siguientes capacidades:

- **Conexión con el backend:** Para cada componente, se hacen solicitudes al backend que permiten obtener los datos procesados desde el servidor, como información sobre las carreras, pilotos, constructores, entre otros.
- **Procesamiento de datos:** Los datos obtenidos del backend se procesan directamente para ajustarlos a la estructura que necesita cada componente y cada visualización. Esto abarca tareas como convertir datos a un formato adecuado para las diferentes funciones como aplicar filtros o la creación de gráficas.

- **Procesamiento de datos del chatbot:** En el caso del chatbot, se muestra la respuesta en función de lo que el Modelo ha generado en el backend. Dependiendo de la *utterance* del usuario, se generará texto de respuesta (explicaciones), botones de decisión (cuando existe la gráfica en la aplicación) o bien una gráfica nueva (cuando la gráfica no existe).

Este diseño asegura una experiencia de usuario fluida y una estructura de desarrollo robusta que permite una fácil integración con el backend y una visualización eficiente de los datos.

## 4.4. Desarrollo de la aplicación

### 4.4.1. Arquitectura de la aplicación desarrollada

La arquitectura de la aplicación, ver Figura 4.11, se ha diseñado para garantizar todos los requerimientos No Funcionales nombrados en la Sección 2.2.3.

- **Frontend (Vue y Javascript):** Encargada de la interacción con los usuarios, desarrollado con VUE para realizar una interfaz dinámica y moderna.
- **Backend (Python):** Responsable de procesar las solicitudes, manejar el chatbot y conectarse con la base de datos.
- **Base de Datos (Postgres):** Almacena toda la información de la aplicación, y se puede acceder a ella a través de consultas SQL.

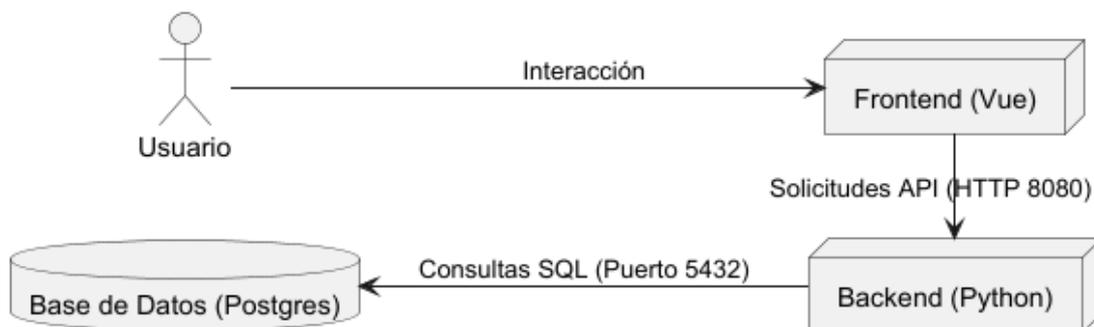


Figura 4.11: Diseño de la Aplicación

En relación con el Frontend, se ha elegido utilizar Vue.js [23], es un framework moderno de Javascript [24] basado en componentes que permite desarrollar aplicaciones de una sola página (SPA) interactivas. Esta opción es perfecta para este proyecto, ya que puede gestionar vistas dinámicas de forma eficiente y evitar duplicación de código delante de PHP [25] y Javascript que es la opción más tradicional que únicamente facilitaría en la creación de una interfaz sencilla y una implementación simple.

Por otro lado, en el Backend, se ha elegido utilizar Python [26] junto con la biblioteca Flask [27] debido a su sintaxis clara y sencilla, lo que facilita la implementación de las llamadas API. Por otro lado, hemos decidido no utilizar Node.js [28], que aunque es conocido por su eficiencia y su facilidad de gestión de múltiples conexiones al mismo tiempo, no cuenta con las herramientas para integrar las funcionalidades del chatbot utilizadas en este proyecto como por ejemplo para NL4DV o la integración directa con OpenAI, ya que estas LLM están diseñadas principalmente para Python. Es por ello que en este proyecto se ha decidido utilizar Python con Flask.

Y por último, en relación con las Bases de Datos que soportará la capa de persistencia de la aplicación se ha escogido PostgreSQL [29] delante de SQLite [30] y SQLAlchemy [31], ya que es sistema de gestión de bases de datos relacional (RDBMS) sólido. PostgreSQL destacada por la habilidad para manejar consultas complejas y ofrece una robustez delante de ambos sistemas de gestión, especialmente en aplicaciones donde se requieren consultas complejas y la capacidad de escalabilidad del proyecto.

En resumen, se eligieron Vue.js, Python con Flask y PostgreSQL como las tecnologías principales para el desarrollo de la aplicación. Estas decisiones garantizan un equilibrio ideal entre simplicidad, eficiencia y escalabilidad de nuestro proyecto.

#### 4.4.2. Backend

Implica el Modelo descrito en la sección 4.2. A continuación se detallan las tecnologías usadas en el chatbot y aspectos de cómo se ha importado la base de datos.

#### Chatbot: Decisiones Tecnológicas

El diseño del chatbot se ha desarrollado mediante las decisiones para garantizar la interpretación de las diferentes entradas descritas en la Sección 4.2.1.

- **Contextos Concretos:** El chatbot tiene en cuenta información contextual para personalizar sus respuestas. Los contextos clave incluyen las siguientes funcionalidades, que también ayudan a la detección de posibles gráficas en la aplicación:
  - **Historial de consultas:** Se guarda el historial de las consultas anteriores para poder así mantener una coherencia en el flujo de conversación. Además, el chatbot guarda el último año agregado a la query. Por ejemplo, el usuario ha solicitado previamente la gráfica del 2022 para Lewis Hamilton, y acto seguido solicita: "¿Y de Max Verstappen?". En este caso el chatbot creará la gráfica para Max Verstappen del 2022.

- **Visualización activa:** El chatbot en caso de recibir la palabra clave ‘**mouse**’ en la frase del usuario, recibirá la información de la gráfica, como el tipo de gráfica, los datos, y así poder adaptar su respuesta al contexto específico del usuario.
- **Textos Especiales:** Para manejar las diferentes consultas, se ha implementado una lógica basada en reglas que implican la detección de palabras clave en la frase introducida por el usuario (*utterance*).
  - La palabra ‘**información**’ se analizará la consulta en forma de texto, eliminando la palabra clave.
  - La palabra ‘**mouse**’ se usa para poder explicar los detalles de las gráficas, en caso de que el usuario mencione esta palabra y se encuentre en una gráfica, el chatbot recibirá esta palabra y la información del mouse.

Estas decisiones permiten al chatbot interpretar las diferentes consultas de forma robusta y eficiente.

En el caso de las posibles salidas, el chatbot diseñado utiliza diferentes herramientas para poder ofrecer las capacidades extendidas en Natural Language Understanding (NLU) y Natural Language Generation (NLG). En la Figura 4.12 se detalla cada una de las tecnologías usadas para elaborar una salida u output del chatbot.

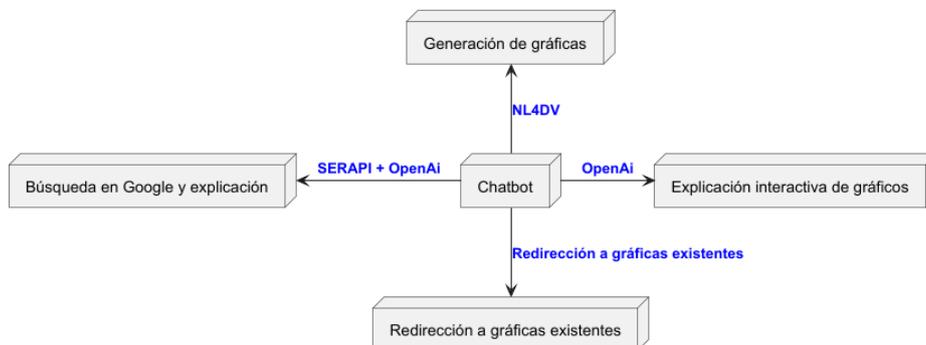


Figura 4.12: Diferentes Funcionalidades del Chatbot y las tecnologías utilizadas.

A continuación se detalla cada bloque mostrado en la Figura 4.12:

- **Generación de Gráficas:** El chatbot interpreta las consultas en lenguaje natural, y genera gráficas dinámicas basadas con el archivo de datos mediante **NL4DV** [32]. Esto permite a los usuarios visualizar información de forma inmediata según su consulta, como se puede ver en la Figura 4.13. Cabe destacar que **NL4DV** solo genera visualizaciones no interactivas, con las que el usuario no puede interactuar directamente. Tampoco **NL4DV** genera textos explicativos sobre las gráficas propuestas.

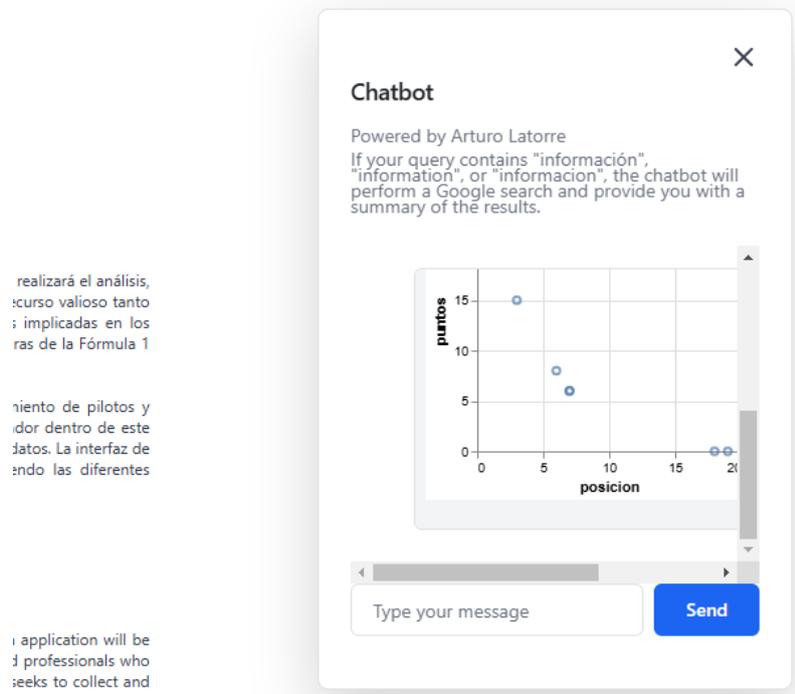


Figura 4.13: Ejemplo de Gráfica Generada por NL4DV

- **Redirección a gráfica:** Cuando un usuario pide una gráfica que se encuentra en la aplicación, el chatbot utiliza un sistema de detección de palabras clave para analizar la consulta y determinar si está asociada a alguna gráfica de la aplicación. Cada gráfica tiene asociadas unas palabras clave. El sistema compara cada palabra con el conjunto de palabras relacionadas con cada gráfica, y se evalúa un porcentaje de coincidencias, si este porcentaje es mayor al de un valor, detecta que se dispone de una gráfica parecida a la pedida, y el chatbot, informa al usuario que puede encontrar una gráfica disponible, como se puede ver en la Figura 4.14. El chatbot da la opción al usuario de poder ser llevado directamente a la gráfica existente. Si el usuario acepta, se le redirigirá de inmediato; en el caso contrario, el chatbot genera una gráfica según la consulta que realizó el usuario, usando **NL4DV** como se ha descrito en la utilidad anterior.

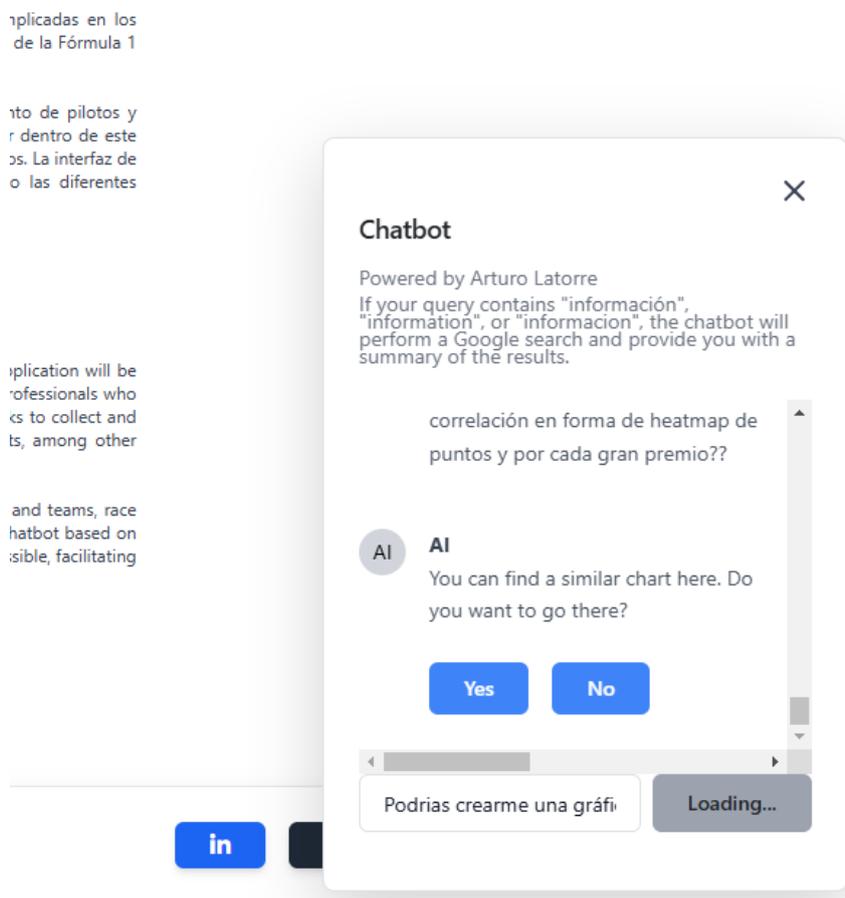


Figura 4.14: Ejemplo de Solicitud de Redirección.

- **Síntesis de búsquedas externas:** El chatbot puede realizar búsquedas externas en Google utilizando **SERAPI** [33] que mediante un GET en el siguiente endpoint `https://serpapi.com/search` junto a la consulta del usuario y la `APIKEY` como parámetros, obtendrá todas las búsquedas en un JSON. Esta funcionalidad se activa cuando el usuario incluye la palabra clave **‘información’** en su query, explicada en las entradas. Así, los resultados obtenidos a través de **SERAPI**, se organizan y se procesan por el chatbot, y junto con **OpenAI** [34], utilizando el modelo `gpt-3.5-turbo` y un máximo de 500 `TOKENS`. Añadiendo el JSON generado en el `PROMPT`, que incluye la consulta a explicar, y se genera una respuesta clara, precisa y detallada, como se puede ver en la Figura 4.15. Además, al acabar la explicación, el chatbot sugiere si el usuario desea ir a alguna sección concreta de la aplicación.

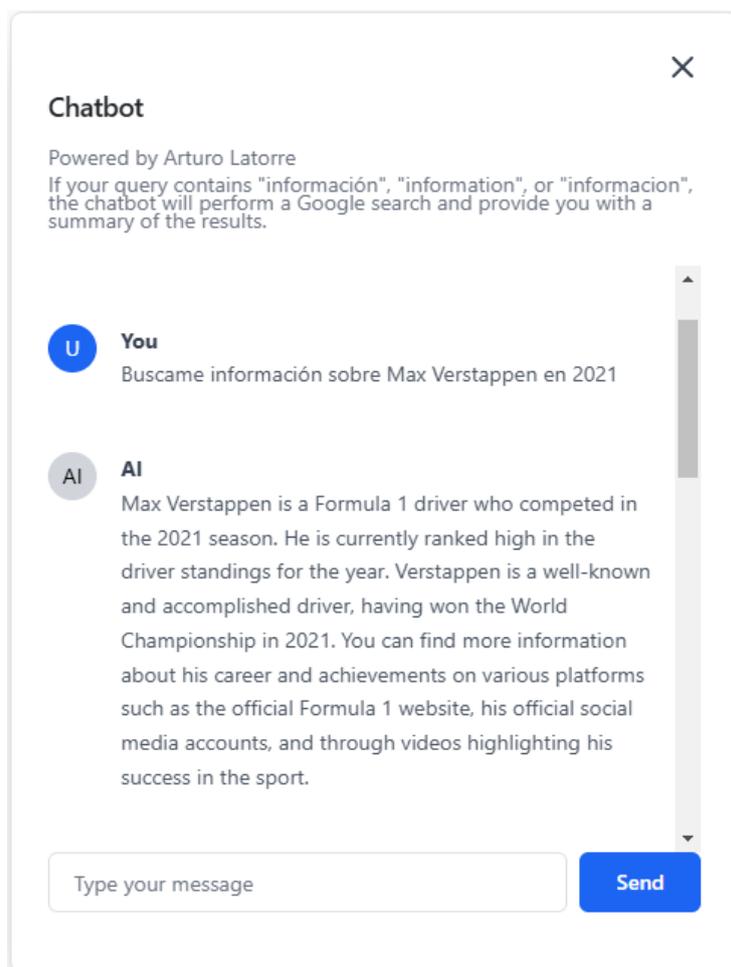


Figura 4.15: Ejemplo de Explicación utilizando SERAPI y OpenAI.

- **Explicación interactiva de la gráfica:** Al posicionar el mouse sobre una gráfica de la aplicación - no una generada por el Chatbot-, y escribir la palabra clave **'mouse'** en la query, en el backend el chatbot de una forma parecida a la anterior, coge los datos de la gráfica obtenidos desde el frontend, y son pasados a **OpenAI** con la instrucción de que es un analista de datos de la Fórmula 1, y que tiene que explicar los datos adjuntados, y un máximo de 200 TOKENS. Es así, que el chatbot retorna al frontend explicaciones sobre los datos representados, ayudando al usuario a interpretar la información de manera efectiva, como se puede ver en la Figura 4.16.

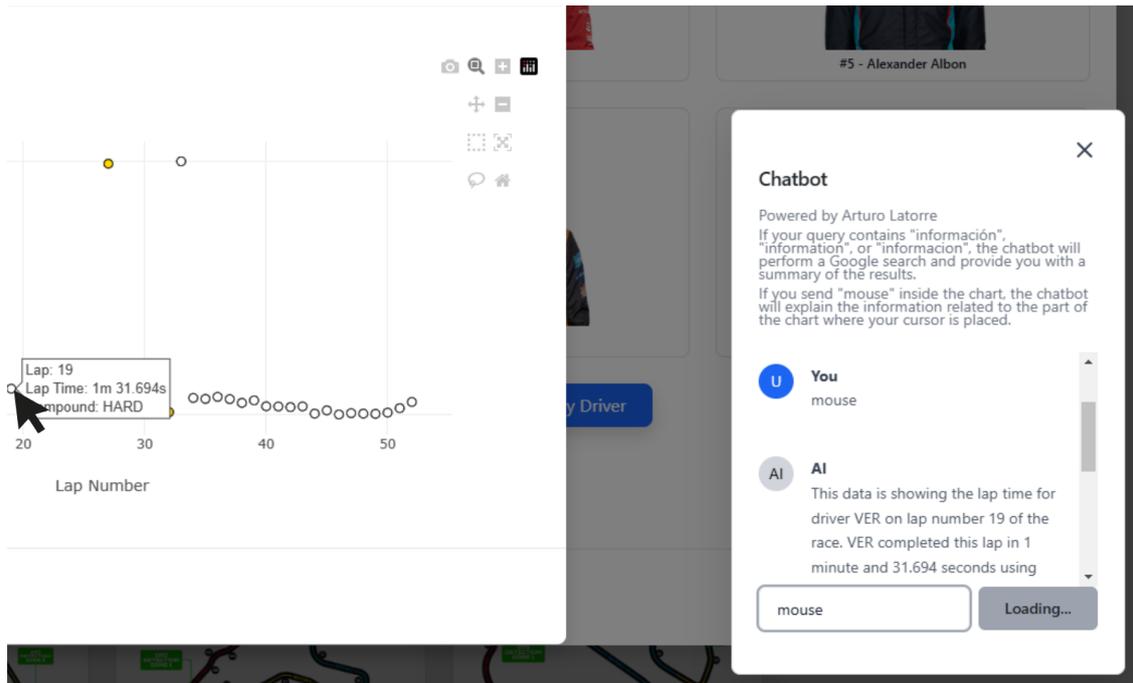


Figura 4.16: Ejemplo de Explicación de la gráfica.

El chatbot se ha diseñado basándose en un sistema de reglas para poder garantizar el cumplimiento de sus propiedades, vistos en la Sección 3.3.1 y utiliza indirectamente la estrategia de RAG en el momento de usar **NL4DV** para generar gráficas nuevas. **NL4DV** necesita los datos contenidos en la base de datos para poder generar las tablas con conocimiento adicional. En Tabla 4.1 se detallan las propiedades que cumple el diseño propuesto.

Chatbot	P1	P2	P3	P4	P5	P6
Evaluación	✓	✓	✓	✓	✓	✓

Tabla 4.1: Evaluación del cumplimiento de los requerimientos del chatbot

En conclusión, el chatbot cumple la mayoría de las propiedades establecidas. Sin embargo, en el caso de la segunda propiedad relacionada con la interpretación de consultas basadas en la información del ratón, se requiere el uso de una palabra clave para poder adjuntar la información requerida, lo cual limita su automatización. Pero a pesar de esta limitación, el chatbot garantiza todas las demás propiedades, asegurando una experiencia interactiva y funcional para todos los usuarios.

## Importación de Datos

Para el origen de los datos, como hemos visto en la Tabla 2.3, la opción más adecuada para el presente TFG es **f1DataR**, ya que su integración en R permite una manipulación más flexible de los datos. En cambio, ninguna de las API analizadas proporciona los datos junto con fotografías de pilotos o escuderías. Es por esta razón que la capacidad de crear archivos en CSV con R es especialmente importante, ya que permitirá agregar estas fotografías y alimentar una base de datos futura de manera eficiente, consultar el Apéndice A.

Además, los datos han sido importados y guardados en un formato adecuado para que el sistema de gestión de bases de datos escogido en la Sección 4.4.1, **PostgreSQL** pueda leer los datos sin necesidad de realizar una limpieza o transformación adicional durante la carga.

Cabe destacar que también **f1DataR** dispone de valores de telemetría, como cambios de marcha o velocidades según ubicación del circuito, y más datos detallados de cada piloto durante la carrera. Sin embargo, por motivos de tiempo de carga de datos y para asegurar una mejor media de respuesta, estos datos no han sido añadidos en la implementación del proyecto.

# Capítulo 5

## Resultados

Los resultados de la aplicación muestran una visión específica del análisis realizado en el Capítulo 3. Siguiendo los objetivos establecidos en los casos de uso de las Figuras 3.3 y 3.4. Al abrir la aplicación se muestra una landing page (Figura 5.1), donde introduce al desarrollador y un resumen de este proyecto. Además, es el punto inicial a todos los casos de uso. En esta sección se detallan los resultados obtenidos en los distintos casos de uso y una primera experiencia prospectiva con usuarios.

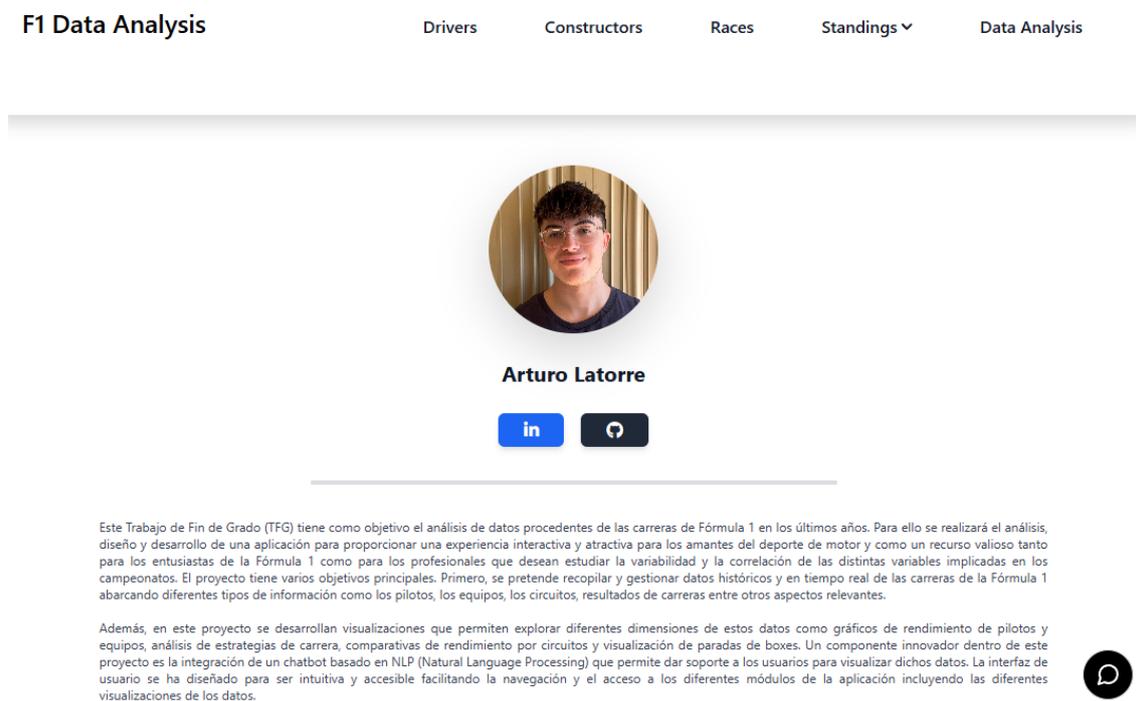


Figura 5.1: Vista de la Landing Page

## 5.1. Resultados Obtenidos

Primero de todo, estudiaremos los resultados obtenidos para el Usuario Casual, ver la Figura 3.1 para seguir con las visualizaciones generadas especialmente para el Usuario Experto.

### 5.1.1. Usuario Casual: Visualizaciones

Como se puede observar, Alba Fernández, nuestra usuaria casual, ha podido visualizar correctamente la lista completa de pilotos, ver Figura 5.2. Los pilotos se organizan en tarjetas individuales, mostrando información clave como su nombre, abreviación, país de origen y bandera asociada. Esta disposición permite una navegación intuitiva para el usuario, (UC1: Mostrar todos los pilotos).

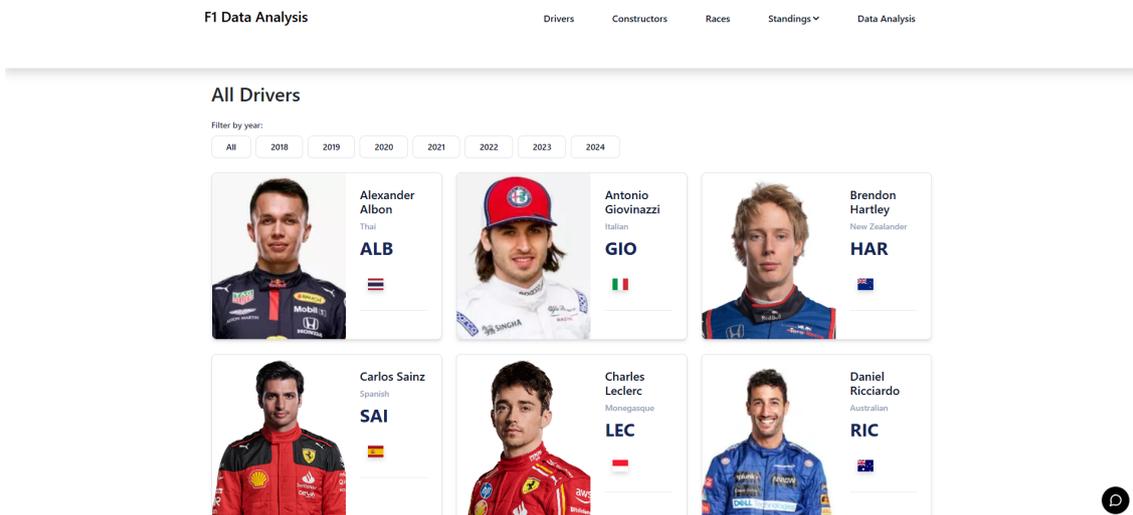


Figura 5.2: Vista de todos los drivers

En esta, Alba utilizó la funcionalidad del filtro por año (UC1.1 Aplicar filtros de un año en específico), seleccionó el año 2023, ver Figura 5.3. La aplicación mostró correctamente los pilotos para esa temporada, organizados de la misma forma que en la Figura 5.2. Esta funcionalidad garantiza que el usuario pueda explorar datos específicos según el año seleccionado, cumpliendo con los objetivos del caso de uso.

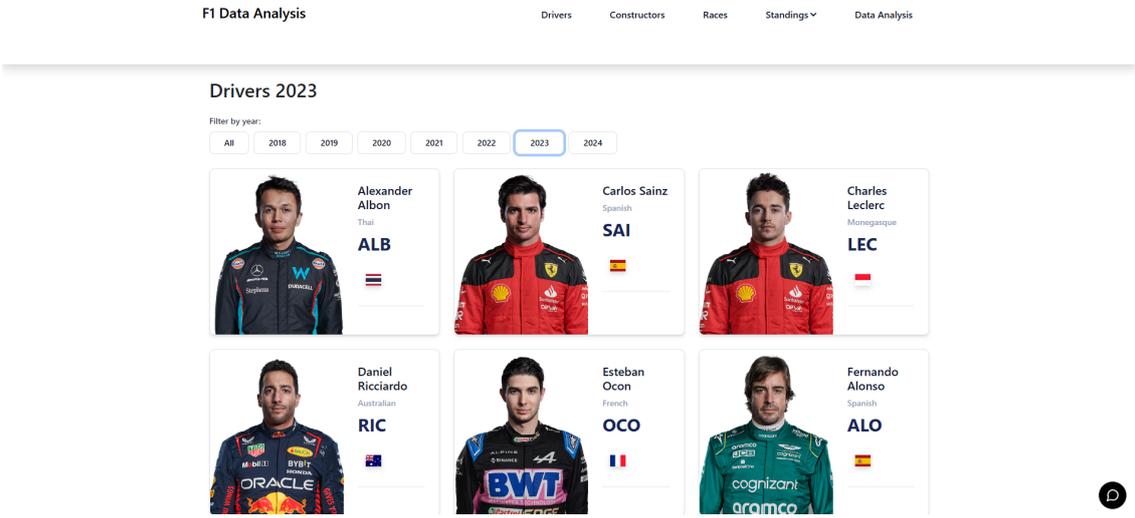


Figura 5.3: Drivers Filtrado en 2023

Asimismo, Alba visualizó correctamente la lista completa de escuderías, ver Figura 5.4. Los constructores se presentan de una forma parecida a los pilotos, pero con un diseño un poco diferente y con la información clave de cada escudería como el nombre, color representativo, país de origen y su respectiva bandera (UC2. Mostrar todos los constructores).

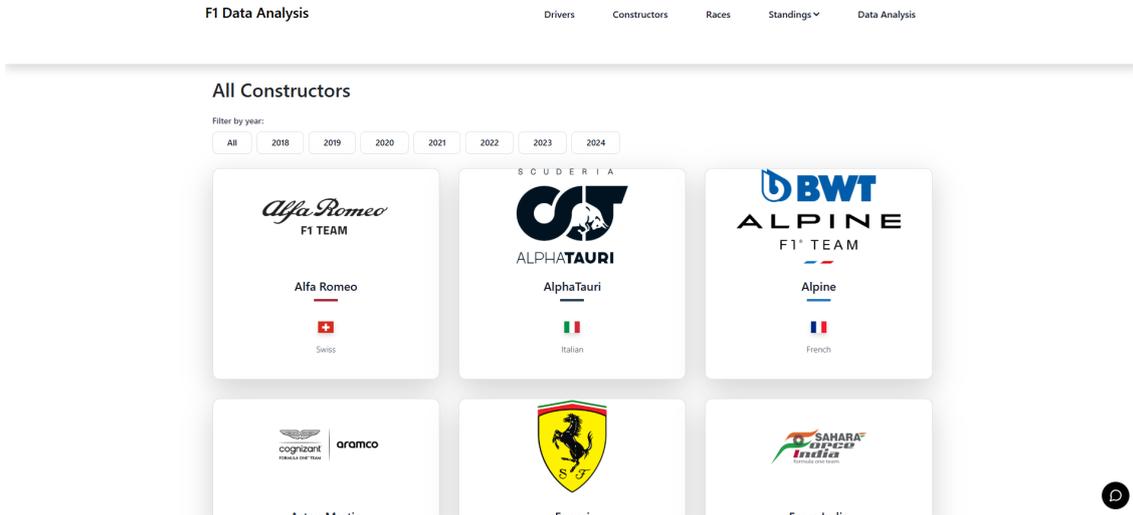


Figura 5.4: Todos los Constructores

Alba puede utilizar el filtro por año (UC2.1 Aplicar filtros de un año en específico) y seleccionó el año 2022, ver la Figura 5.5. La aplicación mostró únicamente las escuderías de esta temporada seleccionada, organizadas de forma similar a la Figura 5.4. Esta funcionalidad permite al usuario explorar información específica de un año determinado, confirmando el correcto cumplimiento del caso de uso.

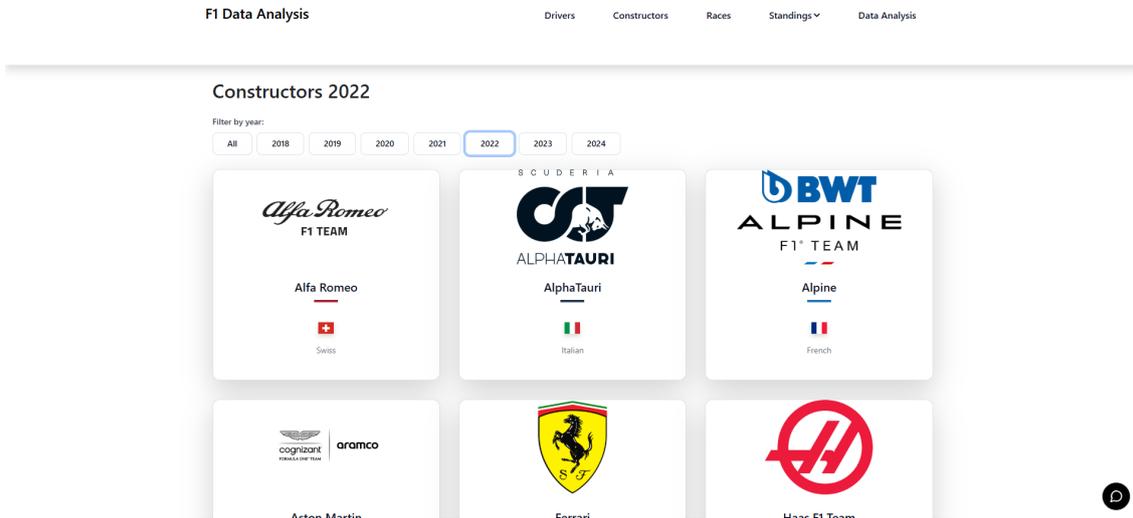


Figura 5.5: Constructores Filtrados en 2022

Alba quiso ver todas las carreras (UC3. Mostrar todas las carreras), ver Figura 5.6. Las carreras se presentan en orden alfabético, donde se muestra en forma de tarjetas, mostrando la información clave de la carrera y el circuito como el nombre de la carrera, del circuito, la fecha, el país donde se organiza, y un botón de información de ese gran premio.

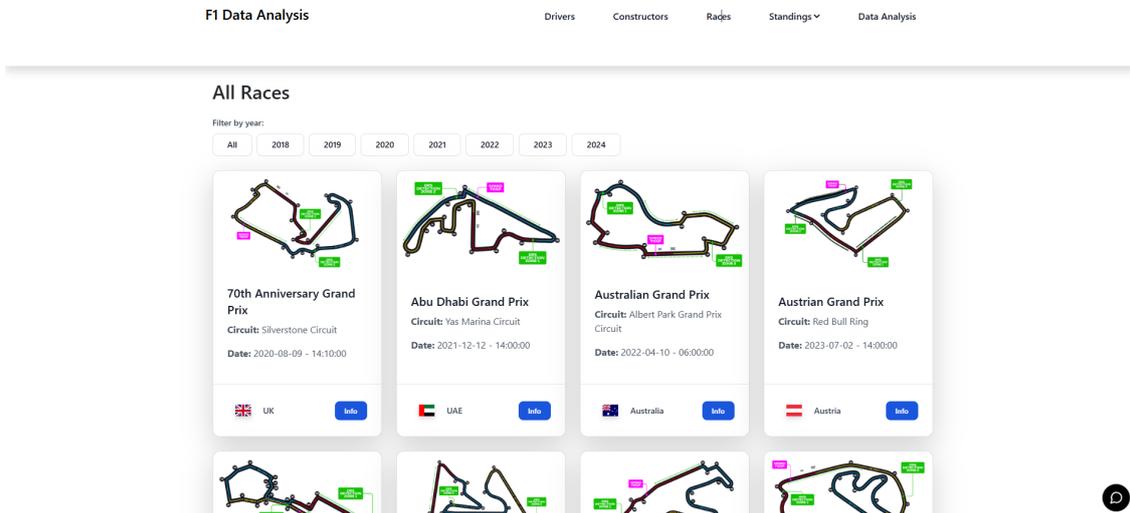


Figura 5.6: Todas las Races

La usuaria utilizó el filtraje por año (UC3.1 Aplicar filtros de un año en específico), seleccionando en el 2023, ver la Figura 5.7. La aplicación enseñó las carreras organizadas de esa temporada, y en orden cronológico. Esta funcionalidad permite a la usuaria explorar información específica de las carreras del año seleccionando.

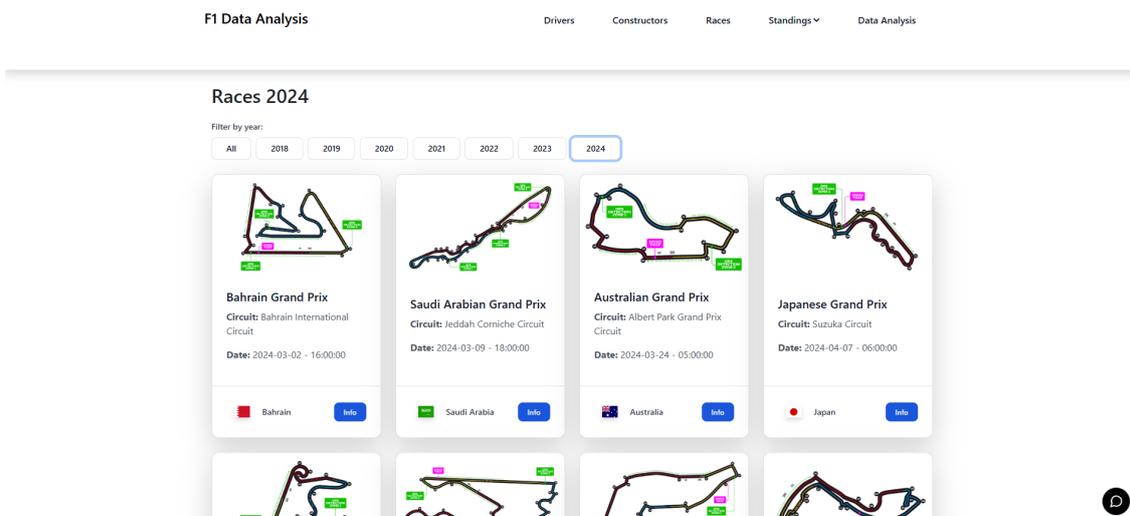


Figura 5.7: Carreras Filtradas en 2024

Para tener una visión más general de las carreras, la usuaria ha decidido pulsar el botón de información de una carrera (UC3.2 Abrir la información de una race), en este caso de Bahrain del 2024, ver Figura 5.8. En ella, se observan las imágenes de los pilotos clasificados en orden de posición, y en páginas de 10 en 10. Y la posibilidad de seleccionar los pilotos, y generar gráficas. Además, Alba puede navegar entre las diferentes páginas de los corredores. Esta funcionalidad permite la creación de diferentes gráficas.

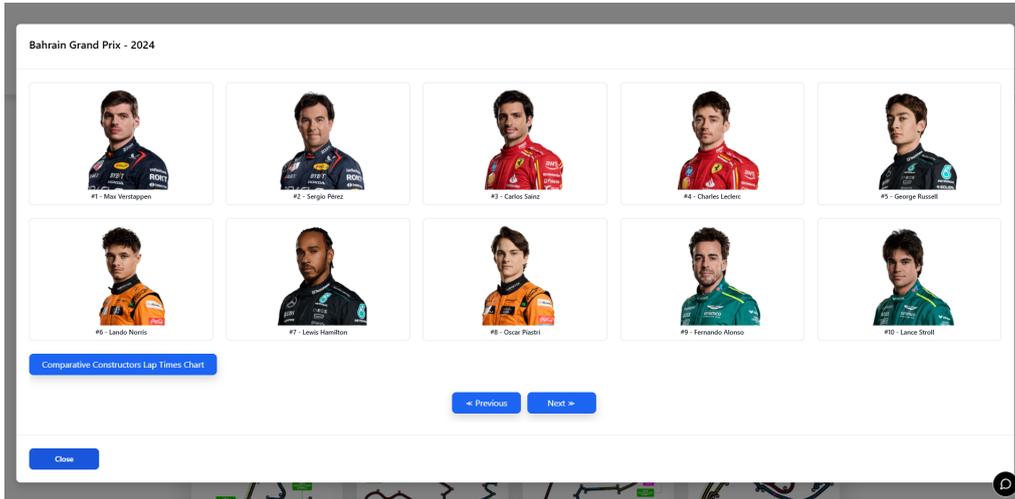


Figura 5.8: Información de Bahrain Grand Prix del 2024

La usuaria generó una gráfica comparativa que muestra los tiempos por vuelta registrados por cada constructor durante la carrera (UC3.2.1 Visualizar la gráfica comparativa de vueltas por constructores), esta gráfica está representada en forma de diagramas de caja y dispersión, ver Figura 5.9, donde se puede identificar patrones como la consistencia de los tiempos por vuelta, el rendimiento promedio de cada equipo, entre otros. Además, la gráfica facilita una evaluación visual clara de la competitividad de las escuderías.

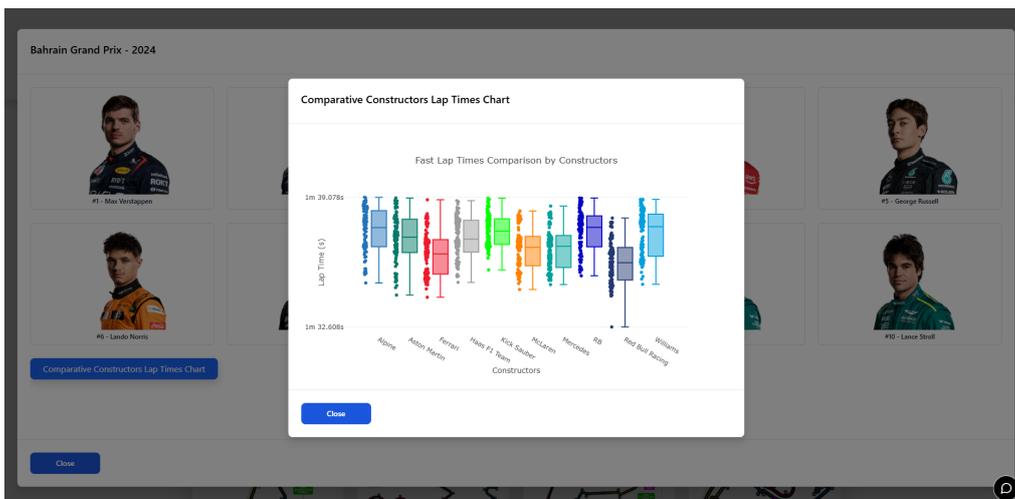


Figura 5.9: Gráfica comparativa de vueltas por constructores en Bahrain Grand Prix 2024.

La usuaria seleccionó varios pilotos para generar una gráfica de posiciones durante la carrera (UC3.2.2 Visualizar la gráfica por posiciones). Esta gráfica muestra cómo cambian las posiciones de los pilotos seleccionados en cada vuelta de la carrera, ver Figura 5.10, permitiendo analizar visualmente el rendimiento y las estrategias a lo largo del evento.

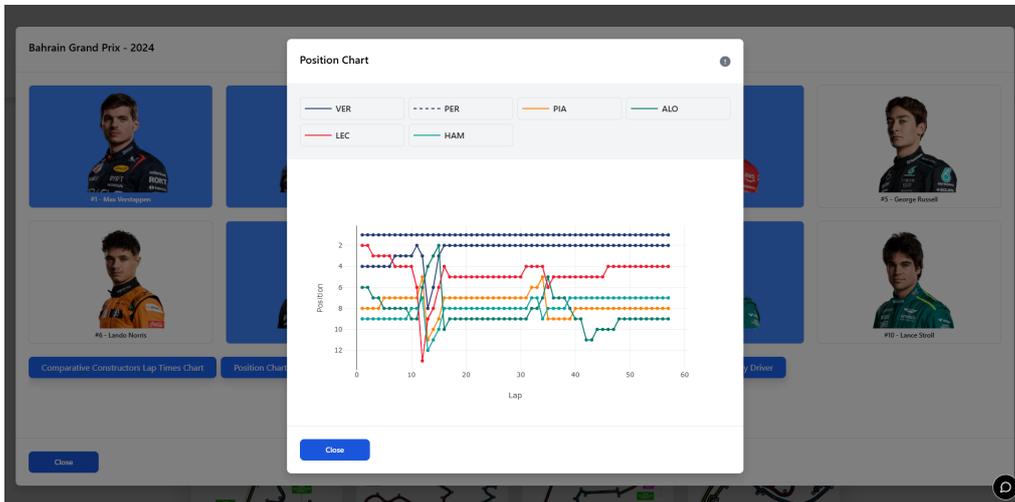


Figura 5.10: Gráfica de posiciones de varios pilotos seleccionados en Bahrain Grand Prix 2024

Al ser una usuaria casual sin mucha experiencia en la aplicación, observo un mensaje de ayuda en la que le indicó cómo proceder para mostrar los pitstops, ver Figura 5.11.

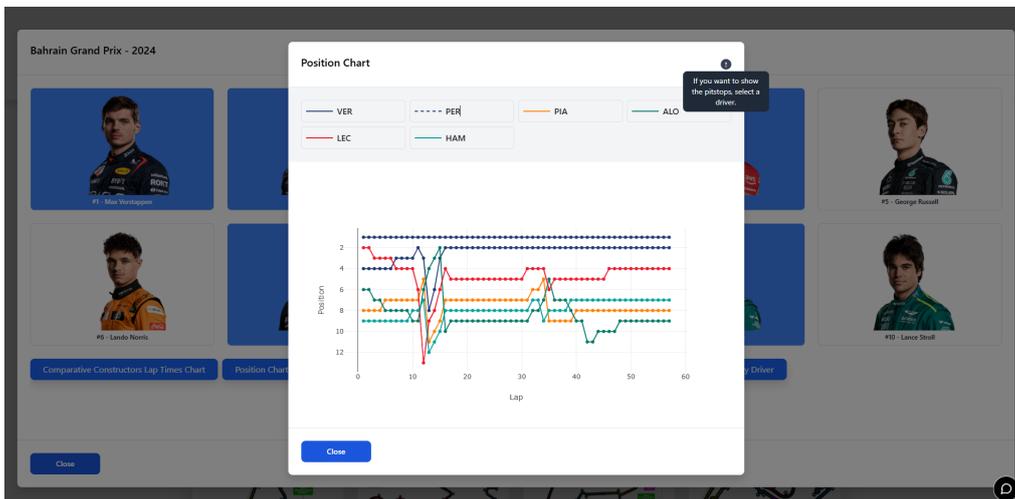


Figura 5.11: Mensaje de ayuda en la gráfica de posiciones

Siguiendo el mensaje, seleccionó un piloto, lo que le activó la funcionalidad de la visualización de pitstops (UC3.2.2.1 Mostrar pitstops), ver Figura 5.12. Los pitstops aparecen representados como marcas verticales, indicando el tiempo de la parada en la vuelta en la que se realizó el pitstop. Esta funcionalidad mejora la experiencia del usuario al ofrecer una guía interactiva, y ayuda en la creación de diferentes estrategias de carrera y a detectar el rendimiento de los pilotos.

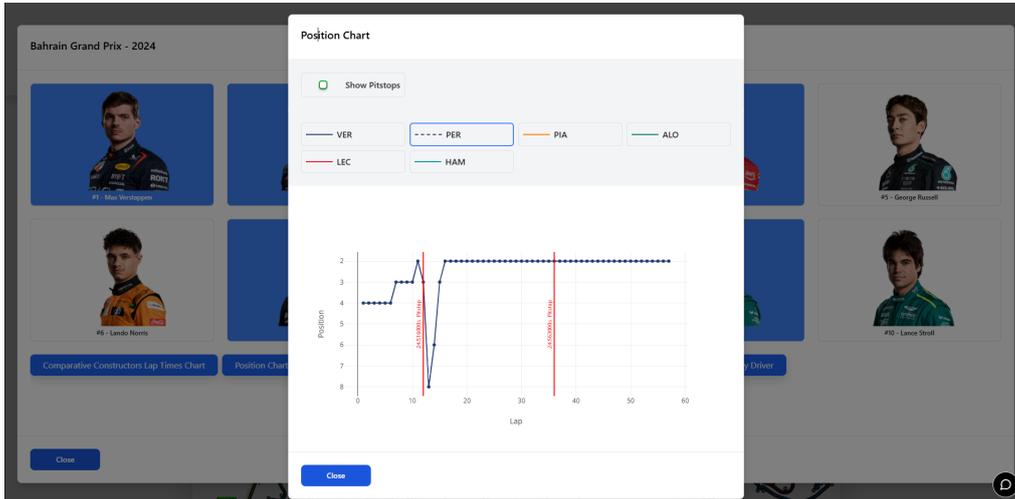


Figura 5.12: Pitstops de la gráfica de posiciones.

La usuaria tiene la posibilidad de generar otras gráficas, y para este caso, seleccionó al piloto Max Verstappen (UC3.2.3 Visualizar la gráfica de tiempos y compuestos por piloto). La aplicación generó una gráfica que muestra los compuestos de neumáticos utilizados y los tiempos por vuelta a lo largo de la carrera, ver Figura 5.13. Esta representación permite identificar cómo influyeron los cambios de compuestos en el rendimiento del piloto y analizar las estrategias utilizadas durante la competencia. Además, la usuaria puede mostrar los pitstops (UC3.2.3.1 Mostrar Pitstops) de la misma manera que se realizó en la Figura 5.12.

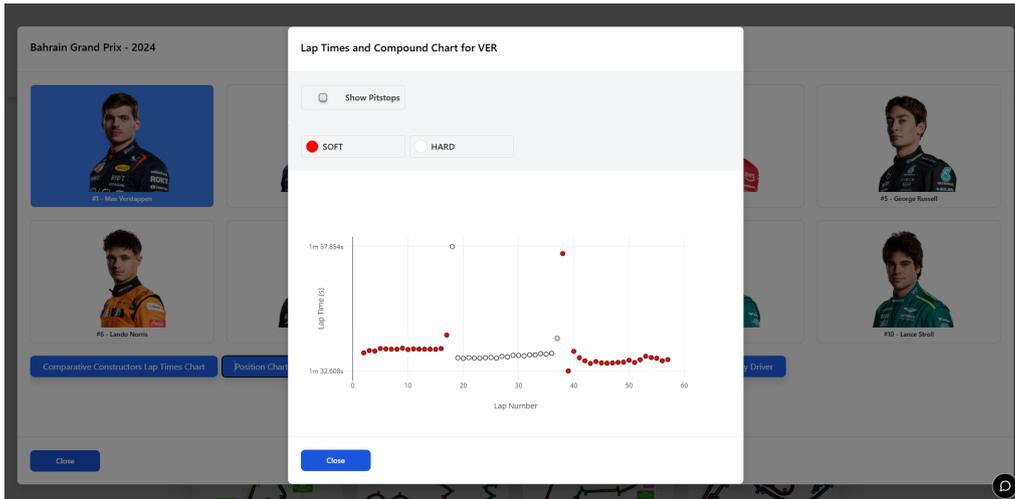


Figura 5.13: Gráfica de compuestos y tiempos por vuelta para Max Verstappen en Bahrain Grand Prix 2024

La usuaria también tiene la funcionalidad para generar una gráfica que muestra los tiempos por vuelta para cada uno de los pilotos seleccionados durante la carrera (UC3.2.4 Visualizar la gráfica de tiempos por piloto). Esta visualización permite analizar el rendimiento de los pilotos a lo largo de la competencia, identificando patrones, estrategias y diferencias significativas en los tiempos por vuelta, ver Figura 5.14.

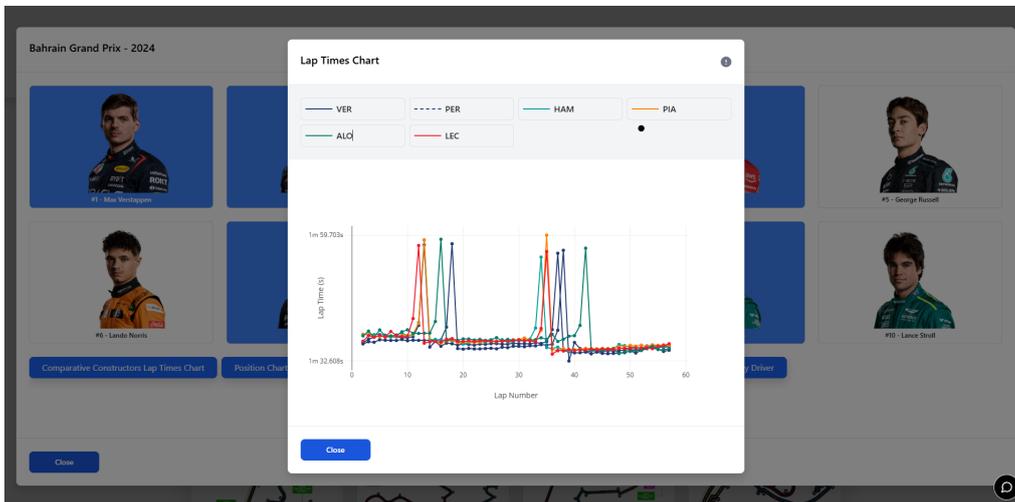


Figura 5.14: Gráfica de tiempos por vuelta para varios pilotos seleccionados en Bahrain Grand Prix 2024.

Al ser una usuaria casual, observó el mismo mensaje de ayuda mostrado en la Figura 5.11, lo que le indicó cómo proceder para habilitar la funcionalidad adicional de los pitstops. Siguiendo estas instrucciones, seleccionó a un piloto en específico, Max Verstappen, para profundizar en el análisis. La aplicación ahora únicamente muestra los diferentes tiempos a lo largo de la carrera para el piloto seleccionado. También muestra la funcionalidad para mostrar los pitstops (UC3.2.4.1 Mostrar Pitstops), donde aparecen de la misma manera que en la Figura 5.12. Esta forma de mostrar los diferentes tiempos de un único piloto, ver Figura 5.15, hace que se pueda ver de una forma más clara para el usuario, y facilitando la comprensión de los datos.

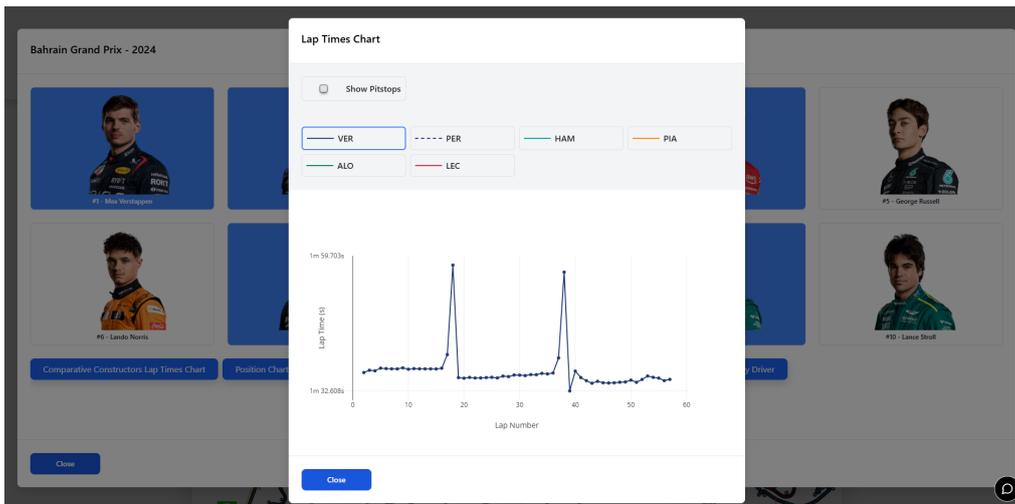


Figura 5.15: Gráfica de tiempos por vuelta para Max Verstappen en Bahrain Grand Prix 2024

Por último, la usuaria tiene la opción de generar una gráfica de distribución de las vueltas rápidas de los pilotos seleccionados (UC3.2.5 Visualizar la gráfica de distribución de las vueltas rápidas). Esta gráfica representa una visualización en forma de violín que permite comprender la dispersión y densidad de los tiempos de vuelta de los pilotos seleccionados, donde cada piloto está identificado tanto por su abreviación como un color de su escudería, ver Figura 5.16. En esta gráfica se pueden observar varios datos matemáticos como valores máximos y mínimos, mediana, entre otros. Esta funcionalidad mejora significativamente la capacidad de análisis al facilitar datos más detallados y profundas de las estrategias y rendimiento de cada piloto.

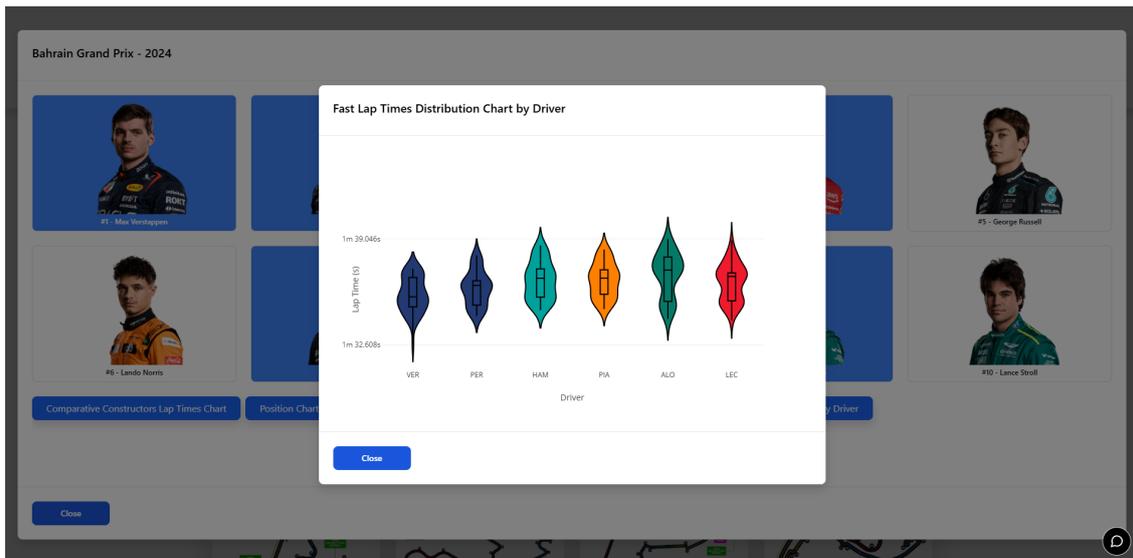


Figura 5.16: Gráfica de distribución de las vueltas rápidas para varios pilotos seleccionados en el Bahrain Grand Prix 2024.

Esta funcionalidad permite a Alba ver de manera clara la posición y los puntos obtenidos en cada Gran Premio, así como los puntos acumulados a lo largo de toda la temporada al pasar el cursor por encima (UC4 Mostrar las tablas de clasificación de los pilotos). La información está organizada de forma intuitiva, ofreciendo una experiencia interactiva que se ajusta a los objetivos del caso de uso, ver Figura 5.17.

Figura 5.17: Driver Standings en 2018.

La usuaria utilizó el filtrado por año (UC4.1 Aplicar filtros de un año en específico), eligiendo el 2023, ver Figura 5.18. La aplicación mostró las clasificaciones de los pilotos y los Grandes Premios organizados de forma cronológica. Esta función enriquece la experiencia de análisis de datos históricos para todos los usuarios.

Figura 5.18: Driver Standings Filtrados en 2023.

Esta funcionalidad es similar a la explicada en la Figura 5.17. Sin embargo, en este caso, la información está organizada por escuderías en lugar de pilotos (UC5 Mostrar las tablas de clasificación de los constructores). Permite al usuario visualizar de manera clara la posición y los puntos obtenidos por cada equipo en cada Gran Premio, así como los puntos acumulados a lo largo de la temporada, ver Figura 5.19. La presentación es igualmente interactiva, mostrando información adicional al pasar el cursor por encima de cada escudería.

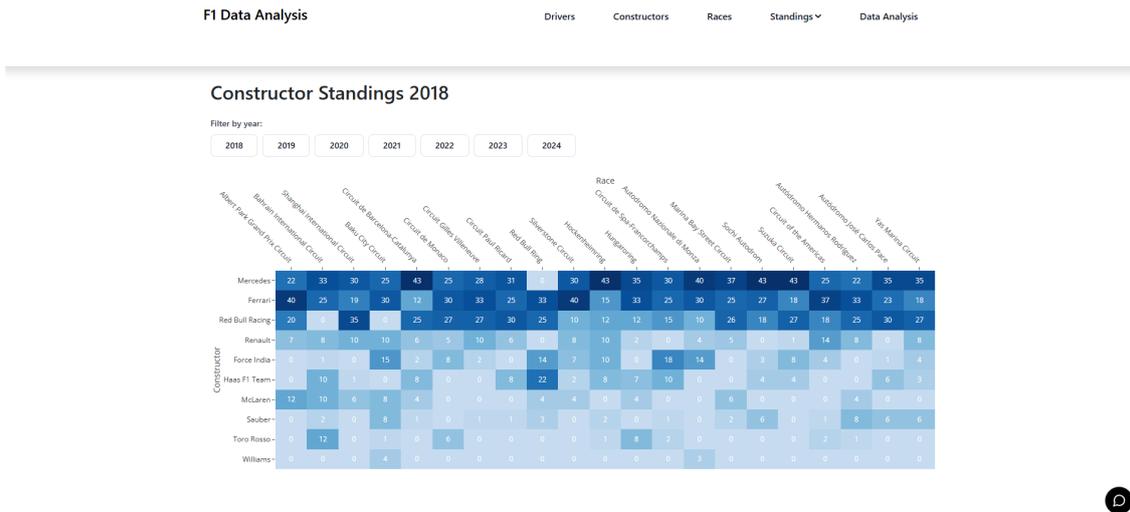


Figura 5.19: Constructor Standings en 2018

Al igual que en la Figura 5.18, la usuaria utilizó la opción de filtrar por año (UC5.1 Aplicar filtros de un año en específico), seleccionando el 2023, ver Figura 5.20. Esta funcionalidad permite enfocar el análisis en una temporada específica, mostrando únicamente los standings de constructores de ese año.

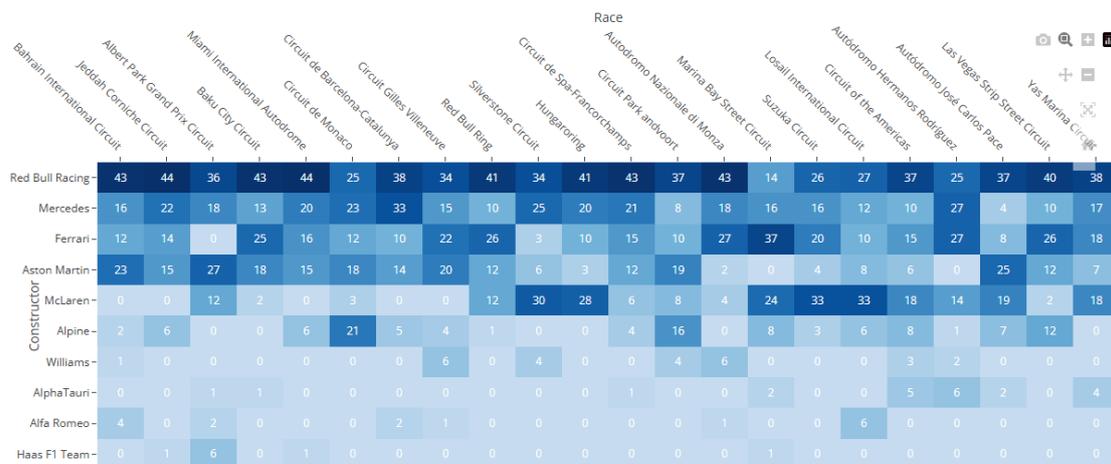


Figura 5.20: Constructor Standings Filtrados en 2023

La usuaria puede abrir el chatbot desde el botón inferior (UC7 Hablar con el chatbot), utilizando esta funcionalidad, ver Figura 5.21, permite interactuar con una herramienta de soporte, proporcionando una experiencia interactiva.

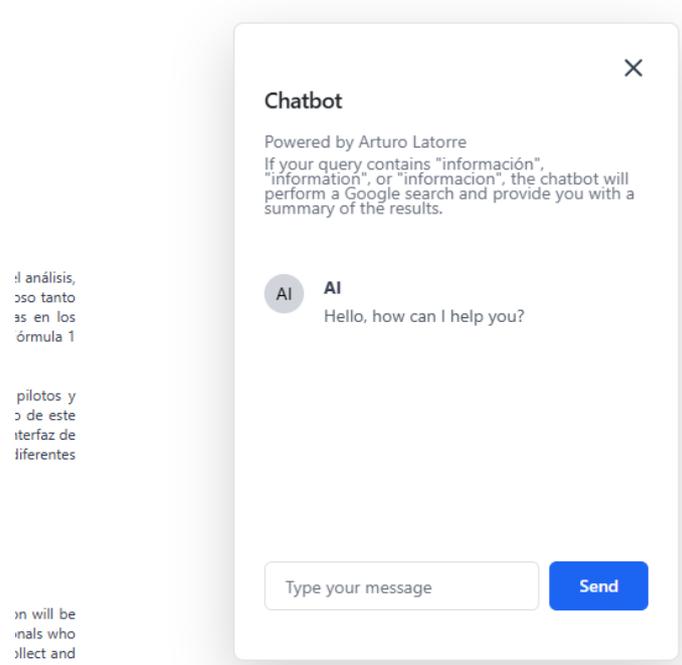


Figura 5.21: Menú Principal con el Overlay del Chatbot abierto

La usuaria puede utilizar el chatbot para solicitar la generación de gráficas específicas (UC7.1 Solicitar gráfica). Esta funcionalidad permite al usuario obtener visualizaciones personalizadas, ver Figura 5.22.

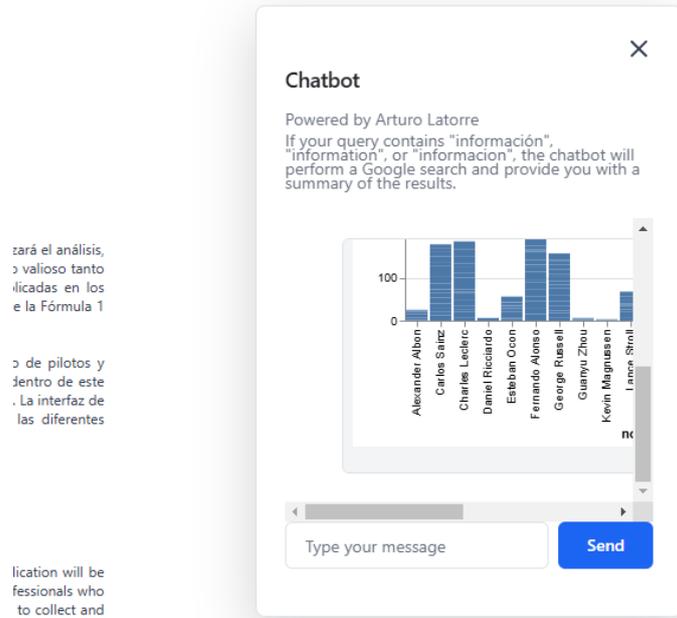


Figura 5.22: Solicitud de gráfica a través del chatbot

La usuaria también puede expandir la gráfica (UC7.1.1 Expandir la gráfica) generada para poder visualizarla de una manera más clara, permitiendo al usuario una experiencia más interactiva, ver Figura 5.23.

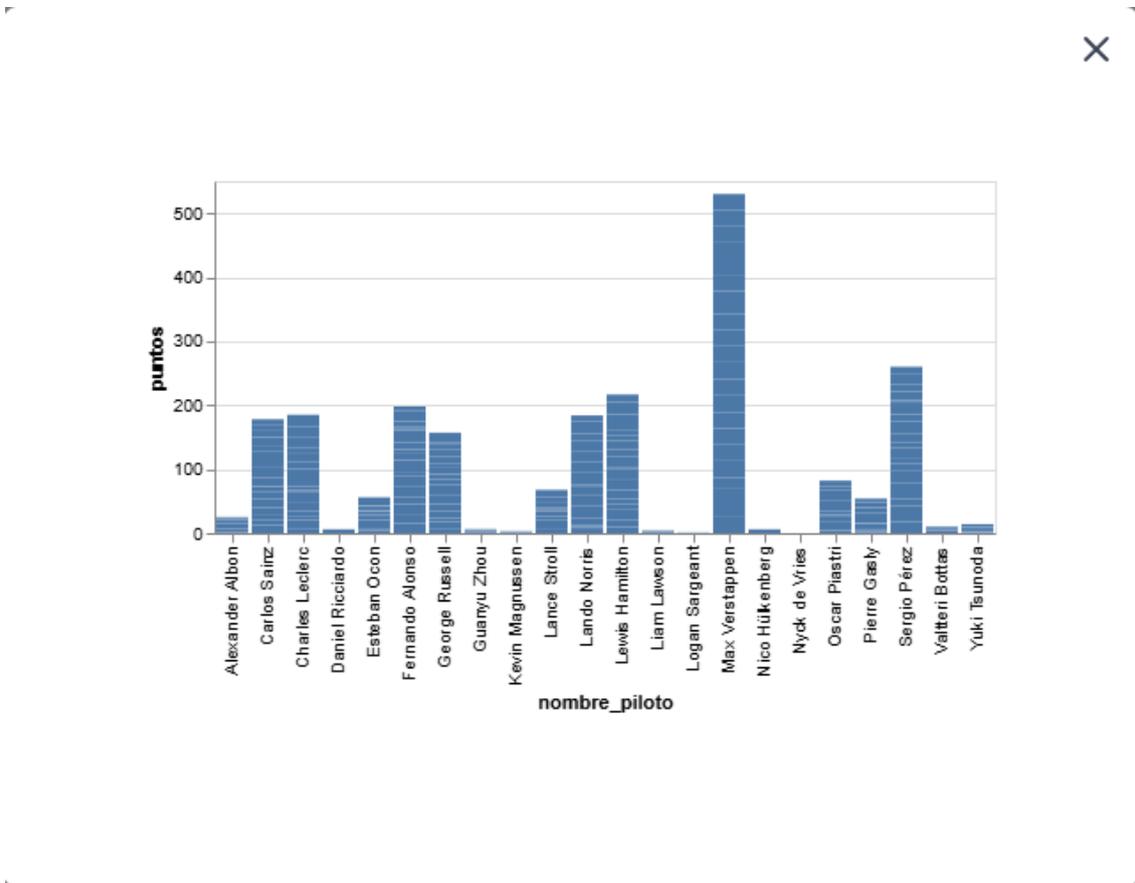


Figura 5.23: Expansión de la gráfica generada

La usuaria puede ser llevada a un apartado específico de la página si el chatbot determina que la gráfica solicitada está disponible en la aplicación (UC7.1.2 Redirección en la página), ver Figura 5.24. La usuaria tiene la opción de aceptar o rechazar esta redirección; si decide aceptarla, será redirigida, y en caso contrario, el chatbot continuará generando la gráfica solicitada.

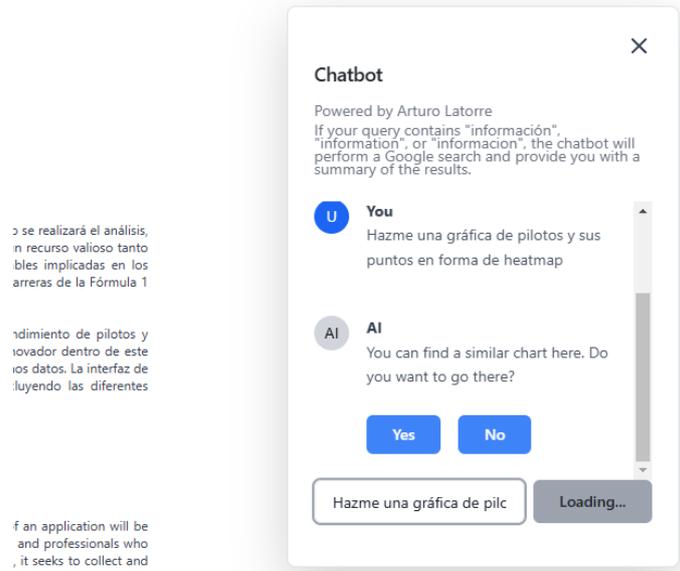


Figura 5.24: Redirección a la gráfica sugerida por el chatbot

La usuaria puede pedir información sobre un tema en específico añadiendo la palabra clave información.<sup>en</sup> su consulta (UC7.2 Solicitar información), ver Figura 5.25.

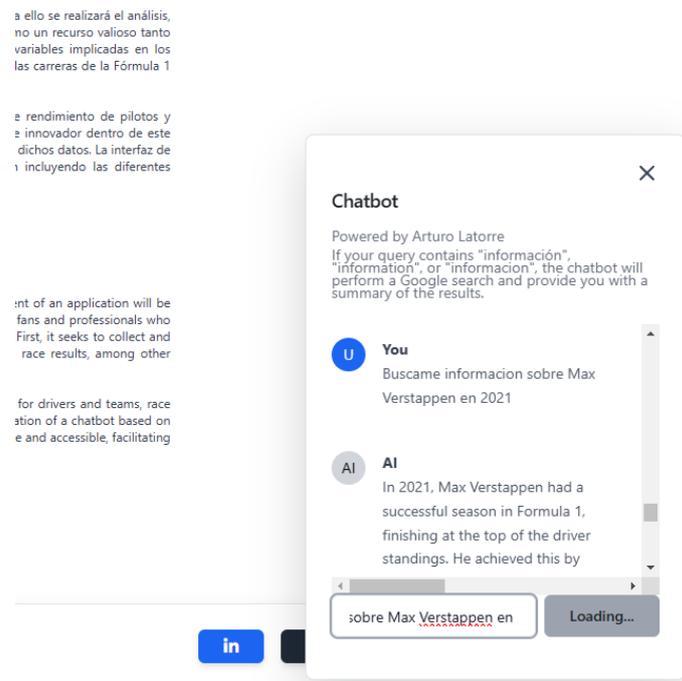


Figura 5.25: Explicación sobre la consulta del usuario

Por último, la usuaria puede pedir información sobre una gráfica que le interese saber información con la palabra clave "mouse".<sup>en</sup> su consulta (UC7.3 Solicitar más detalles de ciertas partes de una gráfica), ver Figura 5.26.

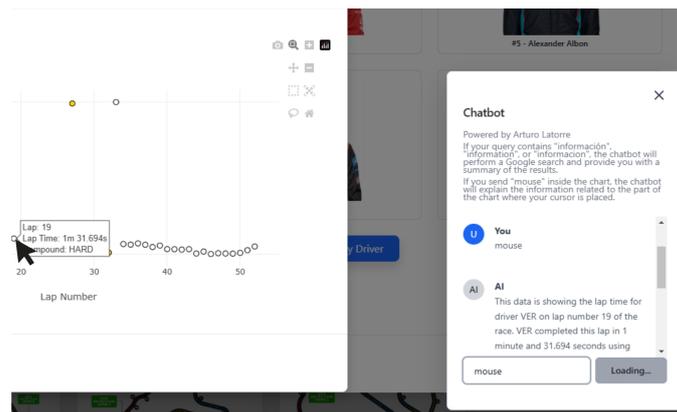


Figura 5.26: Interacción sobre la gráfica utilizando el chatbot.

## 5.1.2. Usuario Experto: Visualizaciones

El Usuario Experto, Donald Alexander, tiene acceso a los mismos casos de uso que el usuario casual, como se puede observar en la Figura 3.4. Sin embargo, lo que diferencia a este usuario es su experiencia y su objetivo dentro de la aplicación. Donald, al ser un usuario avanzado, tiene como función principal realizar análisis de datos más profundos. Por ello, su interacción con la aplicación se enfoca en la sección de **Data Analysis**.

Esta funcionalidad permite al usuario acceder a la creación de una gráfica de correlación de las carreras seleccionadas (UC6: Analizar correlaciones entre datos), organizadas de forma cronológica y filtradas por temporadas, ver Figura 5.27. La información está presentada de manera intuitiva, permitiendo una interacción eficiente y alineada con los objetivos del caso de uso, especialmente para usuarios con experiencia en análisis.

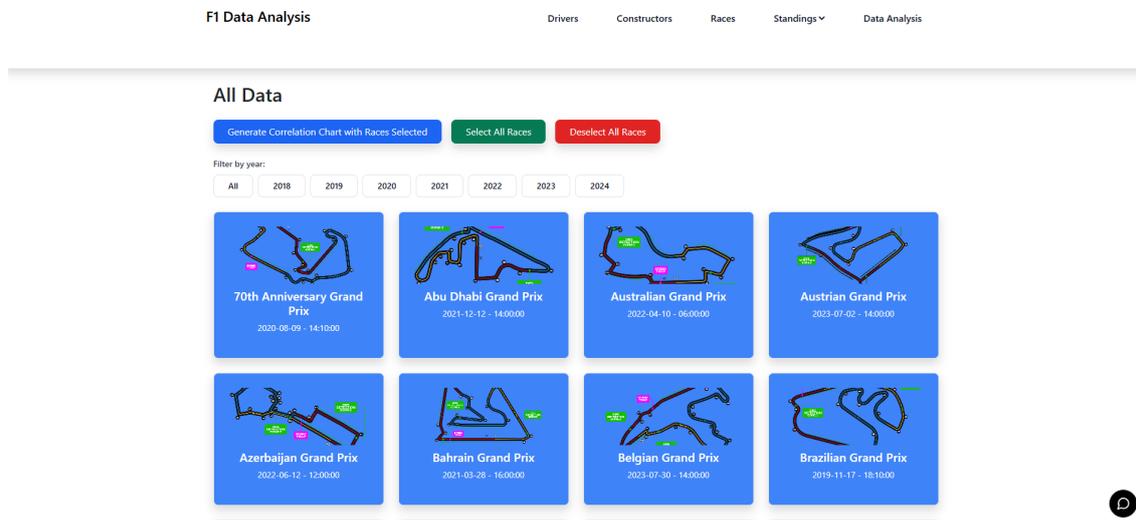


Figura 5.27: Vista de Análisis de Datos

El usuario utilizó la opción de filtrar por año (UC6.1 Aplicar filtros de un año en específico), y seleccionó el año 2021, ver Figura 5.28. Esta función permite al usuario enfocar su análisis en una temporada específica, mostrando únicamente las carreras de ese año.

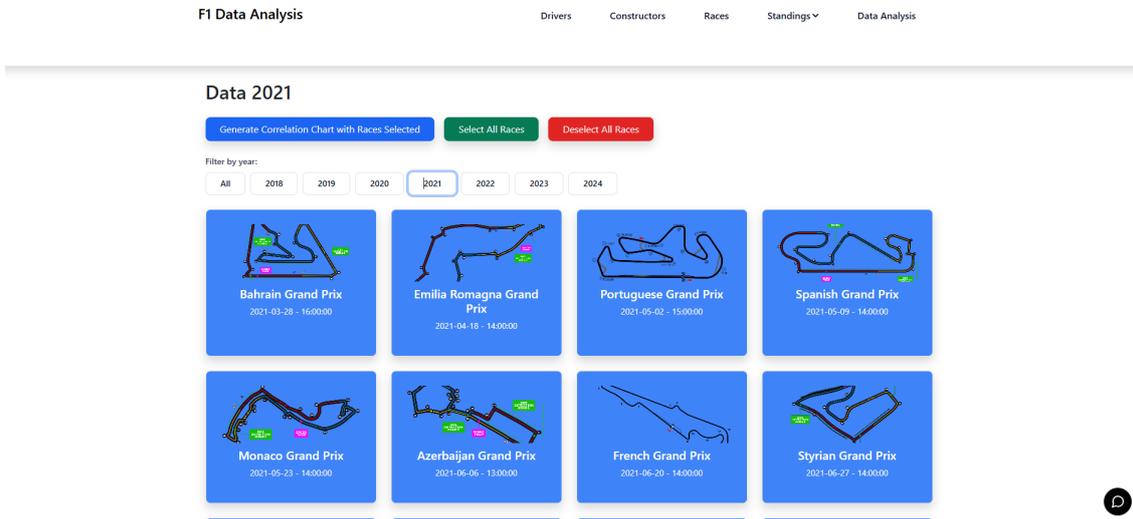


Figura 5.28: Análisis de Datos Filtrados en 2021

El usuario tiene la opción de generar una gráfica de correlación (UC6.2 Visualizar la correlación según las carreras seleccionadas) en caso de tener carreras seleccionadas, ver Figura 5.29. Esta funcionalidad permite visualizar las relaciones entre diferentes variables como tiempos promedio, posiciones, velocidades máximas, entre otros. Los datos se presentan en un mapa de calor, donde las tonalidades reflejan la intensidad de la correlación. En este resultado, se puede ver que las variables relacionadas con los tiempos en los diferentes sectores están altamente correlacionadas, lo que indica que un buen rendimiento, se refleja en los otros sectores. Además, las posiciones finales muestran una correlación negativa con los tiempos promedio, lo que indica que los pilotos con tiempos rápidos tienden a ocupar mejores posiciones en las carreras.

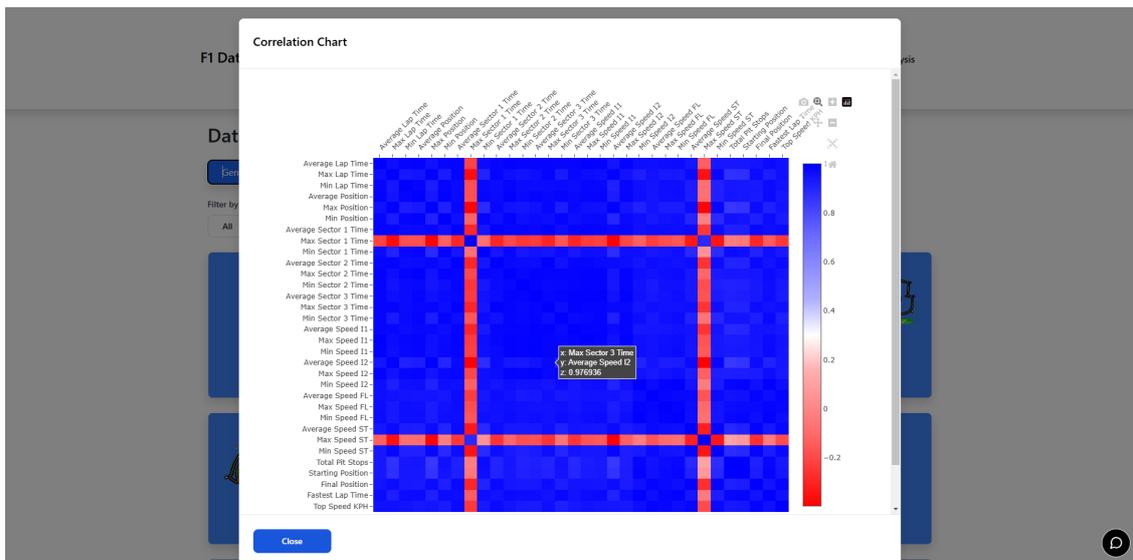


Figura 5.29: Gráfica de Correlación según las carreras seleccionadas del año 2021

El usuario puede utilizar la funcionalidad para seleccionar todas las carreras de manera inmediata (UC6.3 Seleccionar todas las races), este botón permite al usuario ahorrar tiempo al marcar automáticamente todas las carreras para realizar su análisis, facilitando la generación sin necesidad de seleccionarlas todas manualmente, ver Figura 5.30.

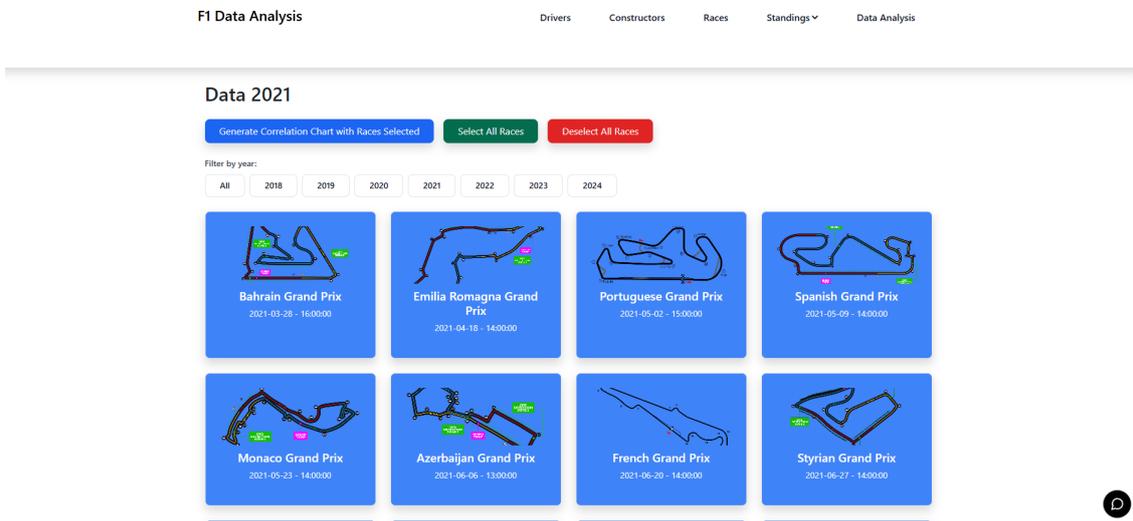


Figura 5.30: Selección de todas las carreras rápidamente para el año 2021

El usuario también tiene la opción de deseleccionar todas las carreras (UC6.4 Deseleccionar todas las races). Este botón permite restablecer las selecciones de forma rápida y eficiente, ver Figura 5.31.

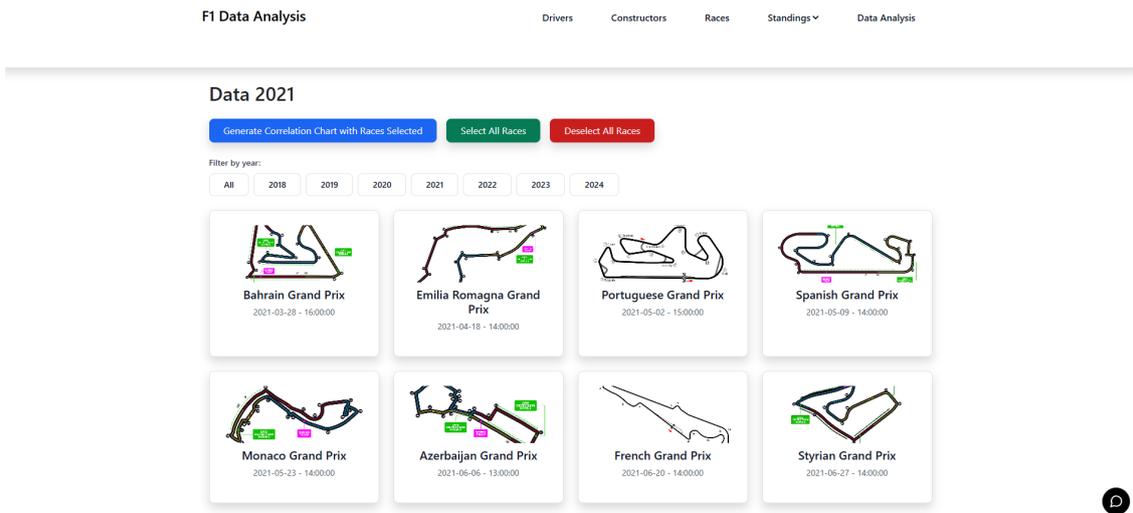


Figura 5.31: Deselección de todas las carreras rápidamente para el año 2021

## 5.2. Experiencia preliminar con usuarios

Para realizar la evaluación de la aplicación, se han seleccionado dos tipos de usuarios diferentes que se corresponden con los usuarios definidos en la Sección 3.1.1. Se han diseñado tareas específicas para ambos roles con el objetivo de evaluar las funcionalidades principales del sistema.

Las tareas son las siguientes:

- Búscas una gráfica sobre Max Verstappen en el Gran Premio de España en el año 2022.
- Puedes pedirle al chatbot que te explique la gráfica?
- Puedes pedirle al chatbot que te cree una gráfica de pilotos en 2022?
- Puedes pedirle información sobre Max Verstappen en 2022?
- Puedes pedirle otra gráfica en forma de heatmap sobre los pilotos en cada gran premio?

Durante la evaluación, se observó que el usuario casual tuvo dificultades iniciales para entender las diferentes secciones de la aplicación. También, encontró problemas para entender que se debían usar palabras clave para pedirle alguna acción concreta al chatbot. Sin embargo, una vez entendió el funcionamiento, se mostró satisfecho con la simplicidad del proceso. Sugirió que sería útil aumentar el tamaño de la fuente y cambiar la colorimetría del menú principal, comentando: "*Estaría bien hacer las letras más grandes para así facilitar a la navegación.*". En general completó las tareas asignadas sin muchas dificultades.

Por otro lado, al usuario experto realizó además la siguiente tarea:

- Puedes analizar los datos de la temporada 2023?

El usuario experto se mostró más cómodo al interactuar con el chatbot, aprovechando su conocimiento previo para realizar alguna consulta más compleja aparte de la tarea solicitada. Sin embargo, indicó que el chatbot en ocasiones no era lo suficientemente preciso al generar gráficas personalizadas, y se debían que dar instrucciones demasiado precisas, y comentó: "*Muy buen chatbot aunque hay que darle explicaciones precisas como cualquier otro chatbot.*" Como mejora, el usuario experto sugirió poder eliminar o añadir alguna que otra variable al hacer el análisis de las variables en cada temporada. También, recomendó añadir la posibilidad de crear gráficas a partir de las vistas de los pilotos o de los constructores.

En conclusión, la evaluación realizada con ambos tipos de usuarios permitió identificar áreas de posibles mejoras en la aplicación. Por un lado, el usuario casual destacó la simplicidad del proceso al entender el proceso, pero señaló dificultades iniciales con la navegación y la necesidad de introducir palabras claves específicas. Por otro lado, el usuario experto valoró la capacidad del chatbot al responder consultas más complejas, indicando que la precisión podría mejorarse, así como la posibilidad de personalizar las variables mostradas en el análisis y la posibilidad de crear gráficas en las vistas de los pilotos o constructores. En general, ambos usuarios, coincidieron que la aplicación resulta útil y cumple con las hipótesis nombradas en la Sección 2.2.1. Estos comentarios subrayan la importancia de llevar a cabo un test de usabilidad más general y formal para acabar de evaluar la aplicación desarrollada.

# Capítulo 6

## Conclusiones y líneas futuras

El proyecto ha logrado todos los casos de uso propuestos, consiguiendo una aplicación de análisis de datos para la Fórmula 1 con la incorporación de un chatbot funcional que hasta ahora no existía en este tipo de aplicación. La aplicación desarrollada permite a los usuarios interactuar de manera intuitiva con los datos mediante consultas en lenguaje natural, ofreciendo explicaciones claras, visualizaciones dinámicas y acceso a información detallada.

En cuanto a los objetivos planteados en la Sección 1, se ha conseguido un diseño e implementación de una arquitectura que es capaz de adaptarse a posibles ampliaciones. También se ha conseguido almacenar todos los datos en una base de datos funcional con un número considerable de variables. Aunque la base de datos de origen de los datos dispone de datos más detallados a nivel de carrera, debido a por el tamaño de datos y tiempo de carga no se han incorporado en el presente proyecto.

Uno de los objetivos principales era la integración de un chatbot para permitir a los usuarios tener una experiencia más eficiente y obtener visualizaciones a partir de consultas en lenguaje natural. Este objetivo se ha cumplido mediante el diseño, desarrollo e integración de un chatbot que en su primera versión, permite a los usuarios ser redirigidos a una gráfica existente, generar gráficos personalizados en función de la consulta, y obtener explicaciones detalladas tanto de las gráficas como de la información que necesite el usuario.

En cuanto al diseño y desarrollo de visualizaciones dinámicas, se ha cumplido a nivel de exploración de datos con gráficas predefinidas. No obstante, algunas interacciones del chatbot con las visualizaciones son limitadas, como lo detallan los comentarios de la experiencia preliminar con usuarios.

En cuanto al diseño y desarrollo de gráficas dinámicas generadas durante la ejecución de la aplicación, se ha cumplido parcialmente, ya que las gráficas generadas mediante NL4DV no alcanzan el nivel de personalización y dinamismo esperado.

Finalmente, se ha garantizado totalmente la accesibilidad a diferentes tipos de usuario, logrando una interfaz clara y funcional tanto para usuarios casuales como expertos en análisis de datos.

De cara a posibles mejoras de la aplicación, se han identificado los puntos según la experiencia preliminar llevada a cabo con usuarios y en las limitaciones del sistema. Una de las líneas futuras identificada es la extracción de las palabras claves utilizadas en la interacción con en el chatbot, para poder tener una experiencia más natural y fluida de conversación.

Finalmente, se contempla la incorporación de los datos de telemetría para ampliar el análisis y proporcionar una visión más completa de las carreras, añadiendo nuevas gráficas que permitan explorar otros tipos de datos de la Fórmula 1 como puede ser mapas de los circuitos con información geolocalizada, así como detalles de interacción en las gráficas actuales como la capacidad de modificar o añadir variables en la gráfica de correlación.

# Bibliografía

- [1] David Hermana. «Historia del Mundial de F1: cuándo se fundó, orígenes y todos los campeones». En: *DAZN Journal online* (2022). Last Access: December 2024. URL: <https://www.dazn.com/es-ES/news/otros/historia-mundial-f1-cuando-fundo-origenes-todos-campeones/36g8s1mltzqd1jswv>
- [2] *Gordon Bennett Cup*. Página oficial de la Gordon Benett Cup. Last Acces: December 2024. URL: <https://gordonbennettcup.racing/>.
- [3] Lluís Llurba. «El origen del mayor campeonato de la historia». En: *RedBull* (2017). Last Access: December 2024. URL: <https://www.redbull.com/es-es/historia-origen-formula-uno-campeonato-mundial>.
- [4] Nacho Losilla. «Cómo funcionan las banderas en la F1». En: *DAZN Journal online* (2024). Last Access: December 2024. URL: <https://www.dazn.com/es-ES/news/otros/como-nuevo-formato-gp-sprint-f1/1enteej3utvh41nc6zmzxun2qq>.
- [5] Mauro Mariani. «¿Cómo funciona la clasificación de Fórmula 1? Los detalles sobre la competencia a una vuelta de la F1». En: *TheSportingNews* (2024). Last Access: December 2024. URL: <https://www.sportingnews.com/es/formula-1/news/como-funciona-la-clasificacion-formula-1-f1/tj9gfmdbwqreas29dtdpwjjd>.
- [6] Jose Carlos de Celis. «El sistema de puntos de la F1: puntuación, posiciones y bonus». En: *Motorsport* (2024). Last Access: December 2024. URL: <https://es.motorsport.com/f1/news/sistema-puntos-posiciones-reparto-formula1/6505476/>.
- [7] Haydn Cobb. «¿Cuántos puntos se reparten si no se completa una carrera?» En: *Motorsport* (2022). Last Access: December 2024. URL: <https://es.motorsport.com/f1/news/formula1-nuevas-reglas-puntuacion-carrera-incompleta/10381974/>.
- [8] Pirelli. «Neumáticos de F1». En: *Pirelli* (2023). Last Access: December 2024. URL: <https://www.pirelli.com/tyres/es-es/motorsport/f1/neumaticos>.
- [9] Albert David Wojtaszczyk. «Los neumáticos de la Fórmula 1: ¿Cómo son y qué tipos hay?» En: *CarAndDriver* (2023). Last Access: December 2024. URL: <https://www.caranddriver.com/es/formula-1/a44302002/neumaticos-f1-tipos-pirelli/>.

- [10] David Plaza. «Qué es un pit stop y cuál es el más rápido de la historia de la Fórmula 1». En: *motor.es* (2023). Last Access: December 2024. URL: <https://www.motor.es/que-es/pit-stop>.
- [11] Eurosport. «La FIA reduce el límite de la velocidad en el 'pit lane' de 100 a 80 kilómetros». En: *Eurosport* (2013). Last Access: December 2024. URL: [https://espanol.eurosport.com/formula-1/la-fia-restringe-el-pit-lane-a-comisarios-y-equipos-y-rebaja-la-velocidad\\_sto3836087/story.shtml](https://espanol.eurosport.com/formula-1/la-fia-restringe-el-pit-lane-a-comisarios-y-equipos-y-rebaja-la-velocidad_sto3836087/story.shtml).
- [12] Ana López Vera. «¿Qué es el DRS en F1, cómo funciona y cuántas veces se puede usar?». En: *AS* (2022). Last Access: December 2024. URL: [https://as.com/motor/2022/03/26/formula\\_1/1648279055\\_467790.html](https://as.com/motor/2022/03/26/formula_1/1648279055_467790.html).
- [13] Israel Zamarrón. «La carrera tecnológica detrás de la Fórmula 1: sensores, datos, 'machine learning'...». En: *Forbes México* (2023). Last Access: December 2024. URL: <https://forbes.com.mx/la-carrera-tecnologica-detras-de-la-formula-1-sensores-datos-machine-learning/#:~:text=Por%20ejemplo%2C%20cada%20autom%C3%B3vil%20de,los%20monoplaza%20hasta%20los%20boxes..>
- [14] Formula 1. *F1 Live Timing*. <https://www.formula1.com/en/timing/f1-live>. Last Access: December 2024. 2024.
- [15] F1 Tempo. *F1 Tempo*. <https://www.f1-tempo.com/>. Last Access: December 2024. 2024.
- [16] Rapit. *Rapit*. <https://rapit.com.br/>. Last Access: December 2024. 2024.
- [17] PitWall. *PitWall*. <https://pitwall.app/>. Last Access: December 2024. 2024.
- [18] F1 Dash. *F1 Dash*. <https://f1-dash.com/>. Last Access: December 2024. 2024.
- [19] Formula1 Dashboard. *Formula1 Dashboard*. <https://formulaldashboard.com/>. Last Access: December 2024. 2024.
- [20] Amazon. «¿Qué es la RAG (generación aumentada por recuperación)?». En: *AWS* (). Last Access: December 2024. URL: <https://aws.amazon.com/es/what-is/retrieval-augmented-generation/>.
- [21] Shane Barker. «5 Steps to Create the Perfect Customer Persona for Your Business». En: *Convert* (2020). Last Access: December 2024. URL: <https://www.convert.com/blog/growth-marketing/5-steps-to-create-the-perfect-customer-persona-for-your-business/>.
- [22] Microsoft. «Modelo-Vista-Modelo de vista (MVVM)». En: *Microsoft* (2024). Last Access: December 2024. URL: <https://learn.microsoft.com/es-es/dotnet/architecture/maui/mvvm>.
- [23] Vue.js. *Vue.js*. <https://vuejs.org/>. Last Access: December 2024. 2014.
- [24] Javascript. *Javascript*. <https://developer.mozilla.org/es/docs/Web/JavaScript>. Last Access: December 2024. 1998.

- [25] PHP. *PHP*. <https://www.php.net/manual/es/intro-what-is.php>. Last Access: December 2024. 2001.
- [26] Python. *Python*. <https://www.python.org/>. Last Access: December 2024. 2001.
- [27] Flask. *Flask*. <https://flask.palletsprojects.com/en/stable/>. Last Access: December 2024. 2010.
- [28] Node.js. *Node.js*. <https://nodejs.org/en>. Last Access: December 2024. 2024.
- [29] PostgreSQL. *PostgreSQL*. <https://www.postgresql.org/>. Last Access: December 2024. 1996.
- [30] SQLite. *SQLite*. <https://www.sqlite.org/>. Last Access: December 2024. 2024.
- [31] SQLAlchemy. *SQLAlchemy*. <https://www.sqlalchemy.org/>. Last Access: December 2024. 2024.
- [32] NL4DV. *NL4DV*. <https://nl4dv.github.io/nl4dv/>. Last Access: December 2024. 2024.
- [33] SERAPI. *SERAPI*. <https://serpapi.com/>. Last Access: December 2024. 2024.
- [34] OpenAI. *OpenAI*. <https://platform.openai.com/docs/api-reference/introduction>. Last Access: December 2024. 2024.

# Apéndice A

## Manual Técnico

### A.1. Estructura del Proyecto

La Figura A.1 muestra la estructura general del proyecto desde la carpeta raíz (Root). En ella, podemos encontrar el `docker-compose.yml` y los diferentes diagramas `.puml` para representar los casos de uso y las arquitecturas de este proyecto.

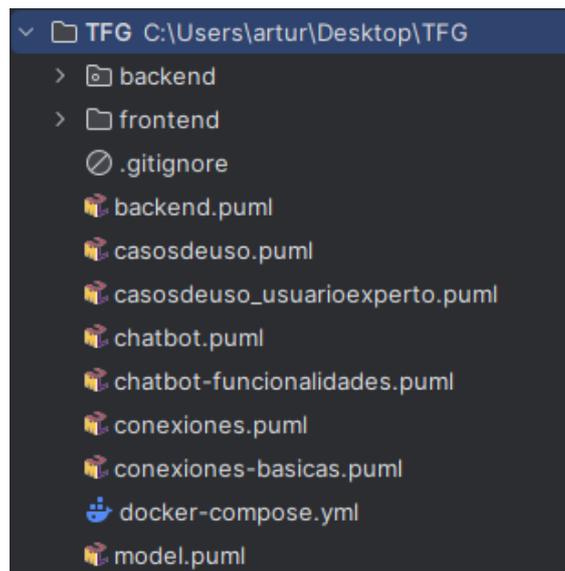


Figura A.1: Estructura del Proyecto

En esta figura se muestran las dos carpetas principales del proyecto:

- **Backend:** Dentro de la carpeta, encontramos todos los archivos para implementar la lógica de la aplicación, tanto las diferentes llamadas API como la lógica del chatbot. Además, encontraremos los archivos para llenar la base de datos, tanto los scripts como los CSV y las diferentes imágenes que necesitamos.
- **Frontend:** La carpeta contiene los archivos relacionados con la interfaz.

Para poder abrir la aplicación, primero de todo, se debe de configurar el archivo del Root `docker-compose.yml` con las API Keys de OpenAI (`OPENAIKEY`) y SerAPI (`SERAPIKEY`), como se muestra en la Figura A.2.

```
backend:
  build:
    context: ./backend
    dockerfile: Dockerfile
  container_name: f1_backend
  ports:
    - "5000:5000"
  environment:
    POSTGRES_USER: "user"
    POSTGRES_PASSWORD: "12345"
    POSTGRES_HOST: "db_postgres"
    POSTGRES_PORT: "5432"
    POSTGRES_DB: "f1-data-importer"
    OPENAIKEY: "tu_openai_key"
    SERAPIKEY: "tu_serapi_key"
  depends_on:
    - postgres
  command: ["sh", "-c", "python main.py"]
```

Figura A.2: Contenedor del Backend donde se muestran las dos variables que se han de modificar.

Después de realizar este cambio, y poder utilizar la aplicación sin ningún problema y poder poblar la base de datos, necesitaremos abrir el proyecto, y llamar al siguiente comando desde el Root: `docker compose up --build`. Este comando abrirá todos los servicios de la aplicación, y cuando estén conectados, es necesario poblar la base de datos.

## A.2. Ajustes en Archivos

Para asegurarse que los datos en la base de datos estén correctamente, se han creado diferentes scripts en **R** que se encuentran en la carpeta `backend/scripts`. Estos scripts generan archivos **CSV** para cada una de las tablas que se mencionan en la Figura 4.4. Los archivos generados se guardan automáticamente en la carpeta `backend/files`, pero es necesario hacer algunos ajustes manuales en ciertos archivos. Aunque actualmente los archivos que se encuentran en la carpeta, ya están modificados.

### A.2.1. Pilotos y Calendario Carreras

En los archivos generados para los **Pilotos** y el **Calendario de Carreras**, es necesario agregar manualmente las URL de las imágenes correspondientes a cada piloto o carrera. Estas URL son las imágenes guardadas dentro de la aplicación, por ejemplo: `"backend/fotos_pilotos/fernando_alonso_2018.jpg"`.

### A.2.2. Constructores

Al igual que en el caso anterior, es necesario adjuntar las URL de las imágenes correspondientes a cada constructor. Además, se debe agregar manualmente los colores en formato hexadecimal de cada escudería, como `"#EF1A2D"` para Ferrari. Estos colores son necesarios para las visualizaciones y poder diferenciar entre distintas escuderías a los pilotos.

### A.3. Como poblar la base de datos

Es necesario utilizar una herramienta de administración como **DBeaver**, ya que necesitamos iniciar las tablas y poblar la base de datos.

1. Conexión a la base de datos con los parámetros de conexión:
  - a) Host: Localhost
  - b) Database: f1-data-importer
  - c) Port: 5432
  - d) Nombre de Usuario: user
  - e) Contraseña: 12345
2. Una vez conectado correctamente, ejecuta el script `backend/init.sql` en DBeaver. Este script crea todas las tablas, anteriormente comentadas en la Figura 4.4.
3. Importa los archivos **CSV** generados anteriormente en cada una de las tablas correspondientes, siguiendo las propiedades de Importación de la Figura A.3.

Name	Value
▼ Properties	
Extension	csv,tsv,txt
Encoding	utf-8
Column delimiter	,
Header position	top
Quote char	"
Escape char	\
<b>NULL value mark</b>	<b>NA</b>
Set empty strings to NULL	[ ]
Date/time format	yyyy-MM-dd[ HH:mm:ss[.SSS]]
Trim whitespaces	[ ]
Timezone ID	
▼ Sampling	
Sample rows count	100
Minimum column length	50
Count length in bytes	[ ]

Figura A.3: Configuración de las diferentes propiedades en Dbeaver