

**CIFP Juan de Colonia**

**Informática y Comunicaciones**

**G.S. Desarrollo de Aplicaciones Web**



**RacingTracker**

Desarrollado por:

**Víctor García Murillo**

19 / 05 / 2025

# Índice de contenidos

[Índice de contenidos 2](#_Toc197893452)

[Índice de figuras 4](#_Toc197893453)

[Índice de tablas 5](#_Toc197893454)

[1. Planificación 6](#_Toc197893455)

[Presentación del problema planteado 6](#_Toc197893456)

[Objetivos 6](#_Toc197893457)

[Requisitos 7](#_Toc197893458)

[Requisitos funcionales 7](#_Toc197893459)

[Requisitos no funcionales 7](#_Toc197893460)

[Recursos 7](#_Toc197893461)

[Recursos de Hardware 7](#_Toc197893462)

[Recursos de Software 7](#_Toc197893463)

[Tecnologías 8](#_Toc197893464)

[Planificación del trabajo 9](#_Toc197893465)

[Planificación económica 9](#_Toc197893466)

[2. Análisis 10](#_Toc197893467)

[Público objetivo 10](#_Toc197893468)

[Nombre de la aplicación 10](#_Toc197893469)

[Diagrama Entidad-Relación 11](#_Toc197893470)

[Diagrama de la aplicación 12](#_Toc197893471)

[Diagrama de casos de uso 14](#_Toc197893472)

[3. Diseño 15](#_Toc197893473)

[Prototipo de la interfaz 15](#_Toc197893474)

[Primeros prototipos 15](#_Toc197893475)

[Prototipos finales 17](#_Toc197893476)

[Diagrama relacional 19](#_Toc197893477)

[Paleta de colores 20](#_Toc197893478)

[Tipografía 21](#_Toc197893479)

[4. Pruebas 22](#_Toc197893480)

[5. Diario de trabajo 23](#_Toc197893481)

[6. Conclusiones 27](#_Toc197893482)

[Objetivos cumplidos 27](#_Toc197893483)

[Propuestas de mejora y ampliaciones futuras 27](#_Toc197893484)

[7. Guía 28](#_Toc197893485)

[Despliegue 28](#_Toc197893486)

[Requisitos previos 28](#_Toc197893487)

[Instalación 28](#_Toc197893488)

[Guía de administración 28](#_Toc197893489)

[Crear pilotos 28](#_Toc197893490)

[Editar pilotos 29](#_Toc197893491)

[Borrar pilotos 30](#_Toc197893492)

[Crear escuderías 31](#_Toc197893493)

[Editar escuderías 32](#_Toc197893494)

[Borrar escuderías 33](#_Toc197893495)

[Crear carreras 34](#_Toc197893496)

[Editar carreras 35](#_Toc197893497)

[Borrar carreras 36](#_Toc197893498)

[Configuración del administrador 36](#_Toc197893499)

[Cerrar sesión 38](#_Toc197893500)

[Guía de usuario 39](#_Toc197893501)

[Ver información general 39](#_Toc197893502)

[Ver información de las temporadas 39](#_Toc197893503)

[Ver información de una temporada 40](#_Toc197893504)

[Ver información de los pilotos 41](#_Toc197893505)

[Ver información de un piloto 41](#_Toc197893506)

[Ver información de las escuderías 42](#_Toc197893507)

[Ver información de una escudería 43](#_Toc197893508)

[Ver información de las carreras 44](#_Toc197893509)

[Ver información de una carrera 44](#_Toc197893510)

[Iniciar sesión 45](#_Toc197893511)

[Abrir repositorio 46](#_Toc197893512)

[8. Bibliografía 47](#_Toc197893513)

# Índice de figuras

[Figura 1.1. Diagrama de Gantt 9](#_Toc197892407)

[Figura 2.1. Diagrama Entidad-Relación de la base de datos 11](#_Toc197892408)

[Figura 2.2. Diagrama de aplicación 12](#_Toc197892409)

[Figura 2.3. Diagrama del repositorio 13](#_Toc197892410)

[Figura 2.4. Diagrama de casos de uso 14](#_Toc197892411)

[Figura 3.1. Boceto de la página de pilotos y su perfil 15](#_Toc197892412)

[Figura 3.2. Boceto de la página de escuderías y su perfil 16](#_Toc197892413)

[Figura 3.3. Boceto del funcionamiento del menú de navegación 17](#_Toc197892414)

[Figura 3.4. Boceto de la página de inicio 18](#_Toc197892415)

[Figura 3.5. Boceto de la página de pilotos 19](#_Toc197892416)

[Figura 3.6. Diagrama relacional de la base de datos 19](#_Toc197892417)

[Figura 3.7. Paleta de colores 20](#_Toc197892418)

[Figura 3.8. Contraste de la paleta de colores 20](file:///C:\Users\Victor\Documents\dev\RacingTracker\documents\memoria.docx#_Toc197892419)

[Figura 3.9. Ejemplo gráfico de las tipografías 21](file:///C:\Users\Victor\Documents\dev\RacingTracker\documents\memoria.docx#_Toc197892420)

# Índice de tablas

[Tabla 1.1. Planificación económica 9](#_Toc197883118)

[Tabla 4.1: Pruebas realizadas. 22](#_Toc197883119)

# Planificación

## Presentación del problema planteado

En las competiciones de motor, el sistema de puntuación predominante se basa exclusivamente en la posición final de los participantes. Este enfoque tradicional asigna puntuaciones fijas a cada puesto sin considerar factores clave como el rendimiento relativo, las expectativas previas o las condiciones particulares de cada piloto o equipo. Como consecuencia, el sistema actual presenta dos deficiencias principales:

* **Falta de reconocimiento al desempeño relativo**: Un piloto o equipo con recursos limitados que logra un resultado modesto podría estar rindiendo a un nivel excepcional en función de sus capacidades, pero el sistema no refleja este mérito.
* **Poca valoración de la superación de expectativas**: Un piloto de un equipo líder que finaliza en una posición alta simplemente cumple con lo esperado, sin recibir reconocimiento adicional si su desempeño supera las proyecciones iniciales.

Estas limitaciones plantean una pregunta fundamental: ¿es realmente equitativo un sistema de puntuación que solo considera la posición final sin evaluar el contexto ni el rendimiento relativo? La respuesta, desde mi perspectiva, es no. Por ello, propongo el desarrollo de un sistema de clasificación más justo y dinámico, capaz de reflejar con mayor precisión el verdadero mérito de cada piloto y equipo.

## Objetivos

* **Desarrollar un sistema eficiente para la gestión de carreras**: Diseñar una aplicación web intuitiva y funcional que facilite la organización y administración de competiciones de motor, optimizando la gestión de participantes, clasificaciones y resultados.
* **Implementar un sistema de puntuación equitativo**: Integrar un modelo basado en TrueSkill (sistema de clasificación desarrollado por Microsoft Research para el servicio de clasificación y matchmaking), que no solo evalúe la posición final de los participantes, sino que también considere factores como el rendimiento esperado y la evolución del desempeño a lo largo del tiempo. Este enfoque permitirá reconocer tanto a los ganadores como a quienes superan sus propios límites y expectativas.
* **Optimizar la eficiencia y experiencia de usuario**: Garantizar el máximo rendimiento de la plataforma mediante una interfaz intuitiva, tiempos de carga reducidos y una navegación fluida. El objetivo es ofrecer una experiencia excepcional tanto para aficionados y competidores como para las organizaciones encargadas de gestionar las carreras.

## Requisitos

### Requisitos funcionales

* **Gestión de competiciones**: Los usuarios con permisos de administrador podrán gestionar pilotos, escuderías y carreras.
* **Visualización de estadísticas**: Todos los usuarios podrán acceder a:
  + Estadísticas detalladas de cada temporada, incluyendo puntos obtenidos por pilotos y equipos en cada carrera.
  + Historial de clasificaciones y las mejores estadísticas de toda la historia tanto de escuderías como de pilotos.
  + Listado de carreras disputadas en cada temporada.
  + Perfiles de pilotos y escuderías con sus respectivas estadísticas.

### Requisitos no funcionales

* Accesible desde dispositivos de escritorio y móviles.
* Disponible en inglés.
* Probada en navegadores basados en Firefox y Chrome.

## Recursos

### Recursos de Hardware

* Entorno de escritorio Windows 11.
* Dispositivo móvil.

### Recursos de Software

* **Visual Studio Code**: Editor de código desarrollado por Microsoft. Incluye características como autocompletado de código, control de versiones con Git integrado, y una gran cantidad de extensiones que mejoran su funcionalidad, como soporte para frameworks y herramientas de desarrollo.
* **Firefox y Chrome**: Navegadores web.
* **AWS**: Plataforma de servicios en la nube que ofrece herramientas para gestionar aplicaciones sin necesidad de infraestructura física.

### Tecnologías

* **NodeJS**: Entorno de ejecución de JavaScript que permite ejecutar código desde el servidor.
* **SQLite**: Sistema de gestión de bases de datos relacional ligero y autónomo, que se utiliza para almacenar datos de manera estructurada en aplicaciones sin la necesidad de requerir un servidor separado para funcionar; los datos se guardan directamente en un archivo en el sistema de archivos del dispositivo.
* **Laravel + React**: Combinación de Laravel, un framework PHP (backend), y React, una biblioteca de JavaScript (frontend). Laravel se encarga de la lógica del servidor, la gestión de bases de datos y la autenticación, mientras que React maneja la interfaz de usuario. Proporciona herramientas y configuraciones prediseñadas para facilitar el desarrollo del frontend, permitiendo una integración eficiente entre ambos.
* **Eloquent**: Sistema ORM de Laravel que permite interactuar con bases de datos usando PHP. Convierte las filas de la base de datos en objetos sin necesidad de escribir SQL.
* **Tailwind CSS**: Framework de CSS de bajo nivel que permite crear diseños mediante clases predefinidas. A diferencia de otros frameworks, Tailwind no ofrece componentes listos para usar, sino que proporciona un conjunto de clases que se pueden combinar.
* **PHP**: Lenguaje de programación diseñado para el desarrollo web. Se ejecuta en el servidor.
* **TypeScript**: Superset de JavaScript que añade tipado estático y características de programación orientada a objetos.
* **TSX**: Extensión de sintaxis que se usa en TypeScript para trabajar con JSX, un formato de sintaxis similar a HTML utilizado en React. TSX permite escribir componentes React dentro de archivos TypeScript.
* **Nginx**: Servidor web que aloja aplicaciones, gestionando las solicitudes HTTP, sirviendo contenido estático y enviando las solicitudes PHP a PHP-FPM para procesar el código de Laravel.

## Planificación del trabajo

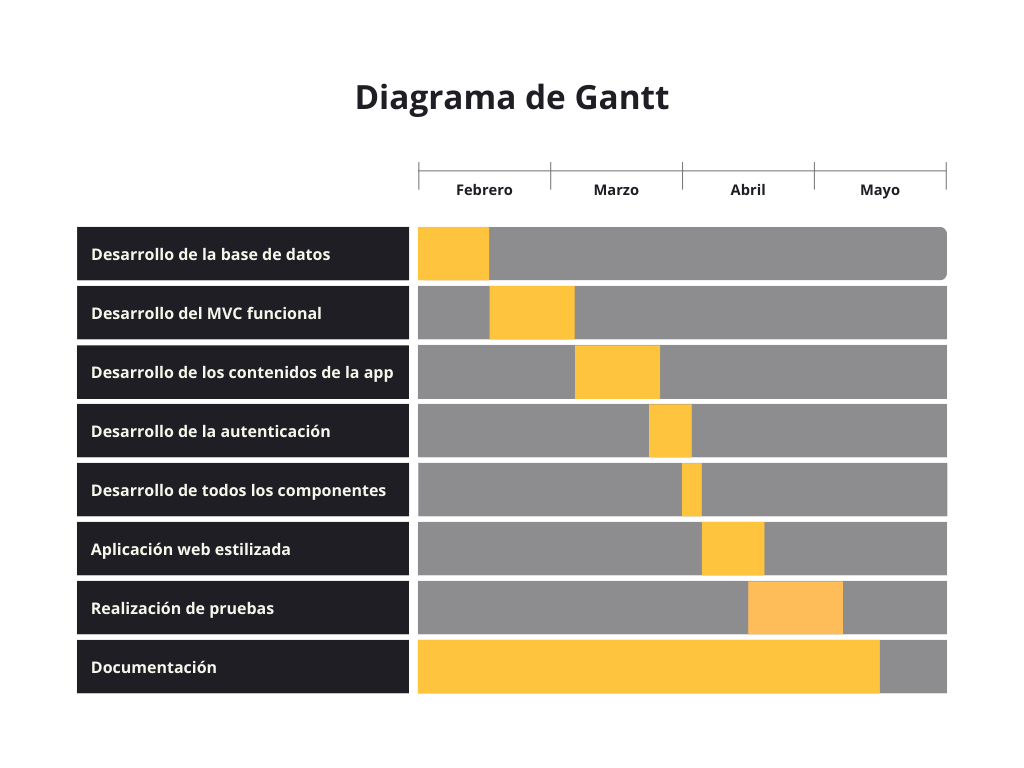


Figura 1.1. Diagrama de Gantt

## Planificación económica

|  |  |  |
| --- | --- | --- |
| Concepto | Detalle / Supuesto | Coste estimado |
| **Ordenador personal** | PC de gama media | 1000,00 € |
| **Programador 30h** | Freelance a 20 €/h | 600,00 € |
| **Gasto eléctrico** 30h | Consumo 100W a 0,25 €/kWh | 0,75 € |
| **Programas** | Licencias básicas | 30,00 € |
| **Dominio** | Dominio común anual | 15,00 € |
| AWS EC2 + VPC | EC2 (tráfico básico 720h/mes) | 35,00 € |
| Hosting DNS | 0,50 €/mes | 0,50 € |
| ****Total**** | | **1.681,25 €** |

Tabla 1.1. Planificación económica

# Análisis

## Público objetivo

Esta aplicación web está dirigida principalmente a empresas que organizan y gestionan eventos de motor, ya que la plataforma está diseñada para ser implementada en sus sitios web.

En segundo lugar, está orientada a aficionados que desean seguir el desempeño de sus pilotos o equipos favoritos, así como a pilotos y equipos que usan la plataforma para ver su puntuación, analizar su rendimiento y compararse con otros competidores.

## Nombre de la aplicación

El nombre de la aplicación, RacingTracker, es simple y clara, reflejando las principales funcionalidades de la aplicación, la gestión y consulta de información relacionada con carreras.

## Diagrama Entidad-Relación

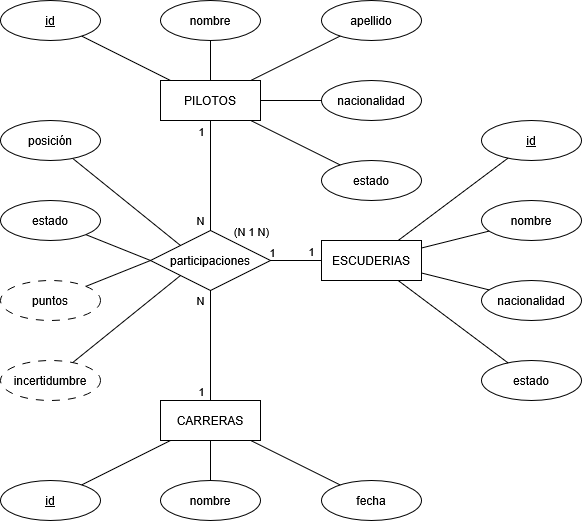


Figura 2.1. Diagrama Entidad-Relación de la base de datos

## Diagrama de la aplicación

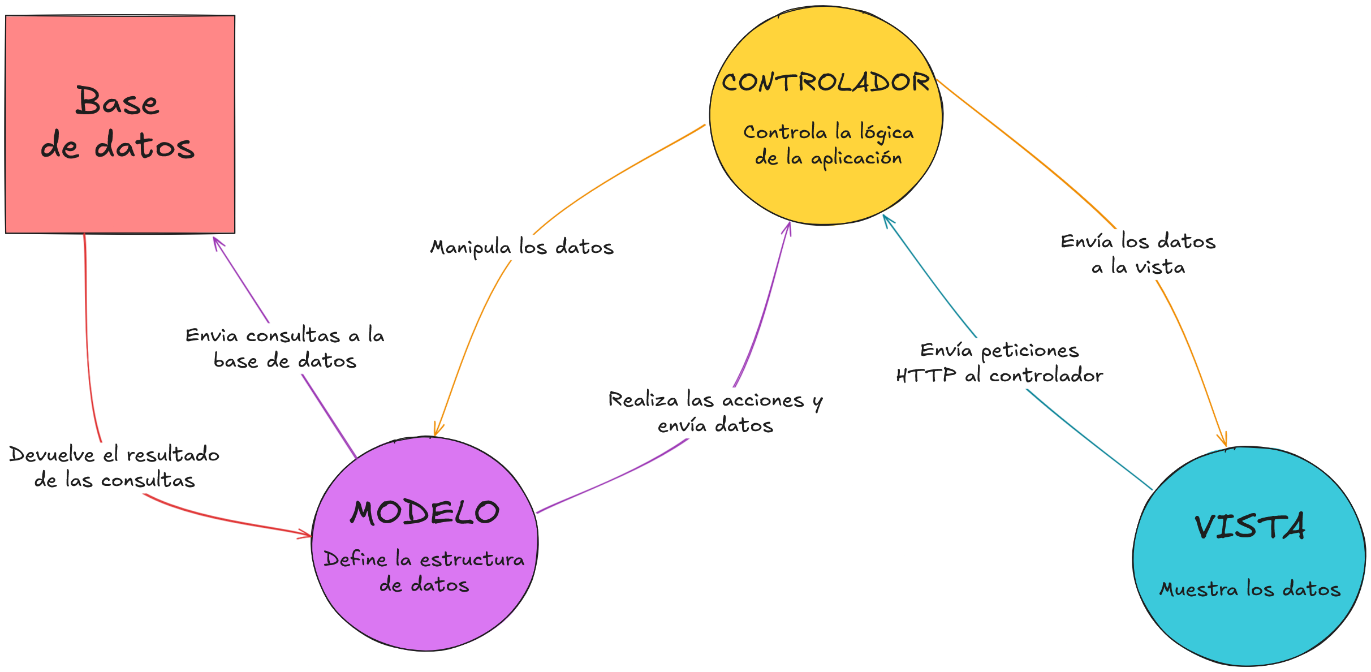


Figura 2.2. Diagrama de aplicación

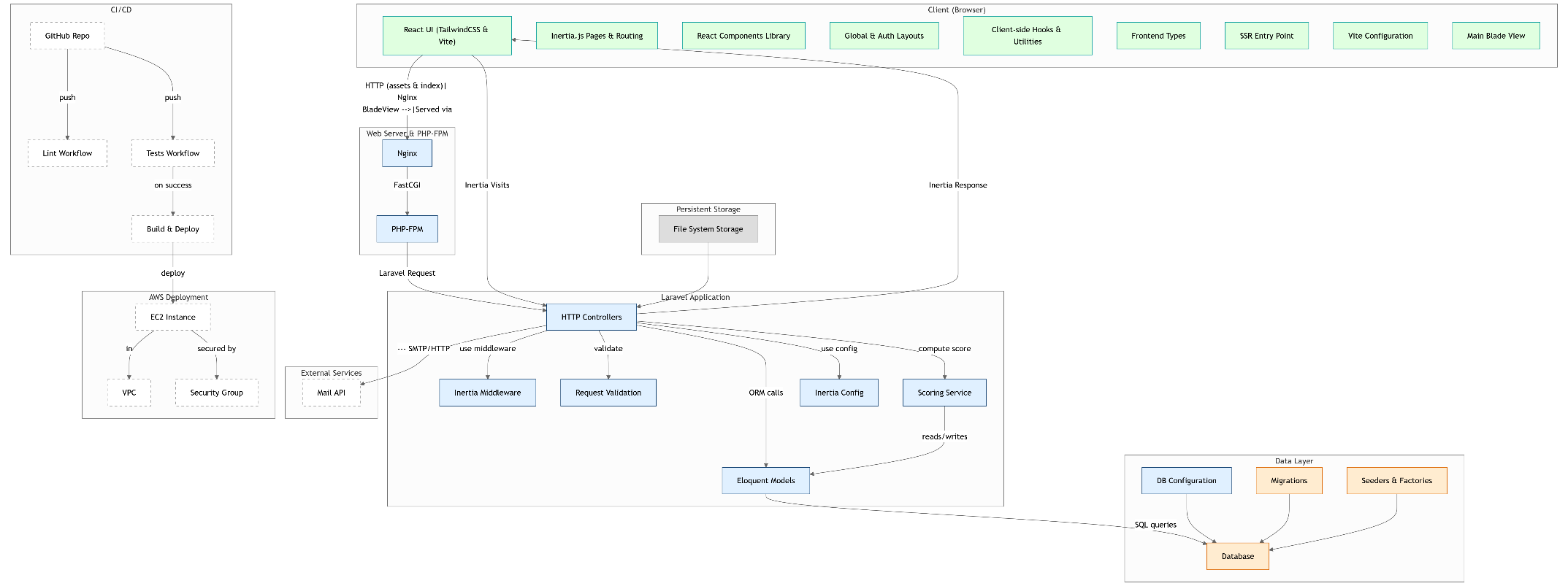


Figura 2.3. Diagrama del repositorio

## Diagrama de casos de uso

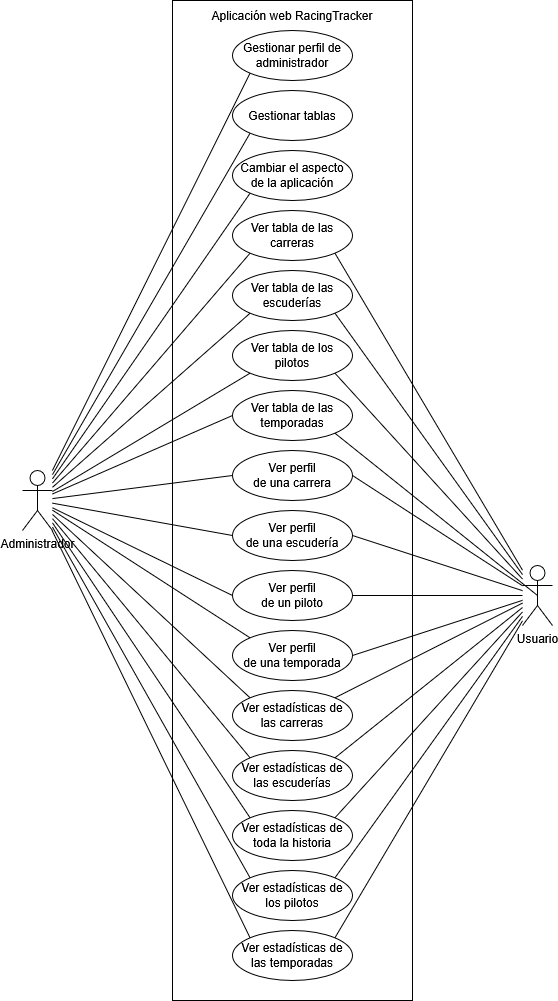


Figura 2.4. Diagrama de casos de uso

# Diseño

## Prototipo de la interfaz

### Primeros prototipos

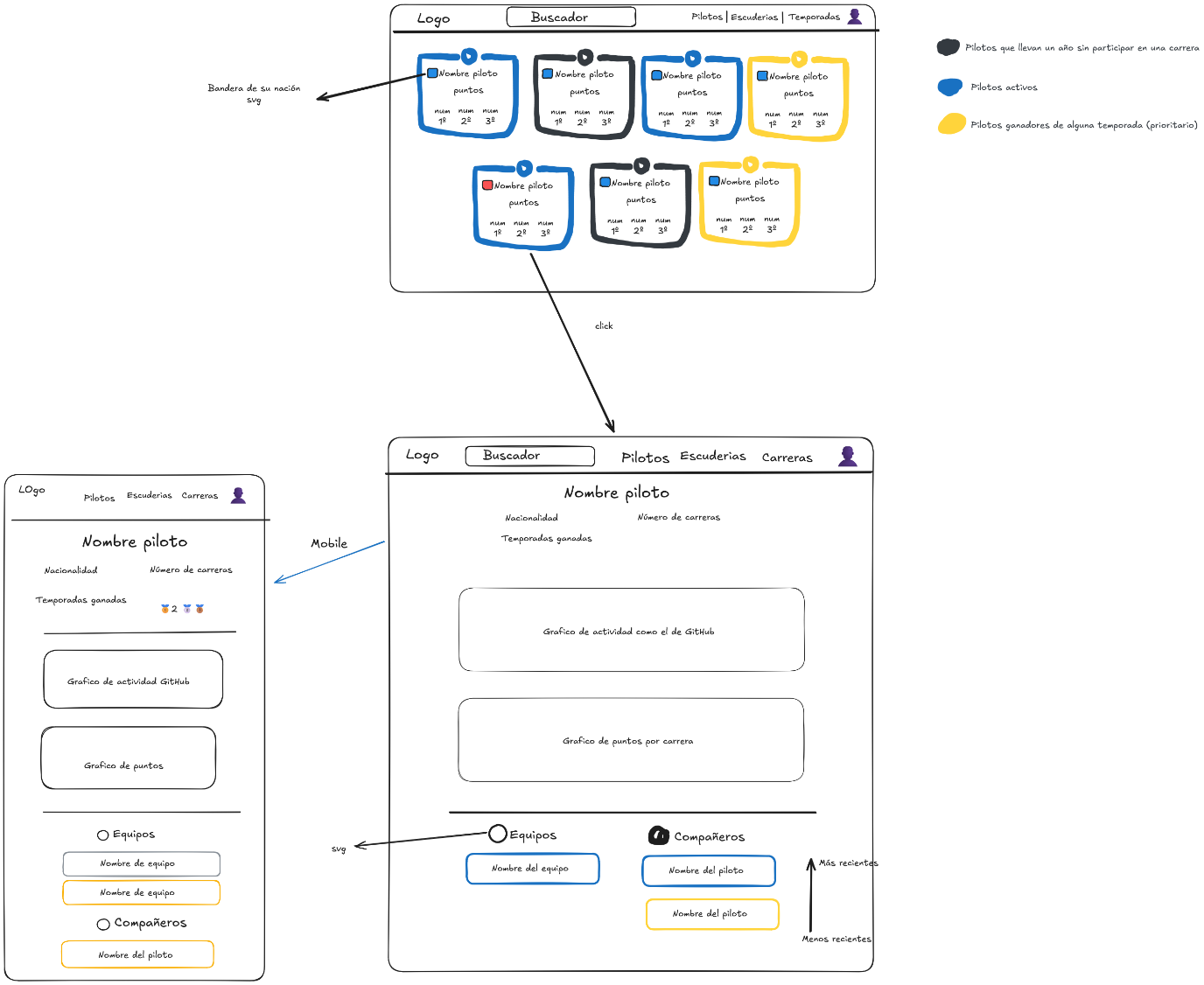


Figura 3.1. Boceto de la página de pilotos y su perfil

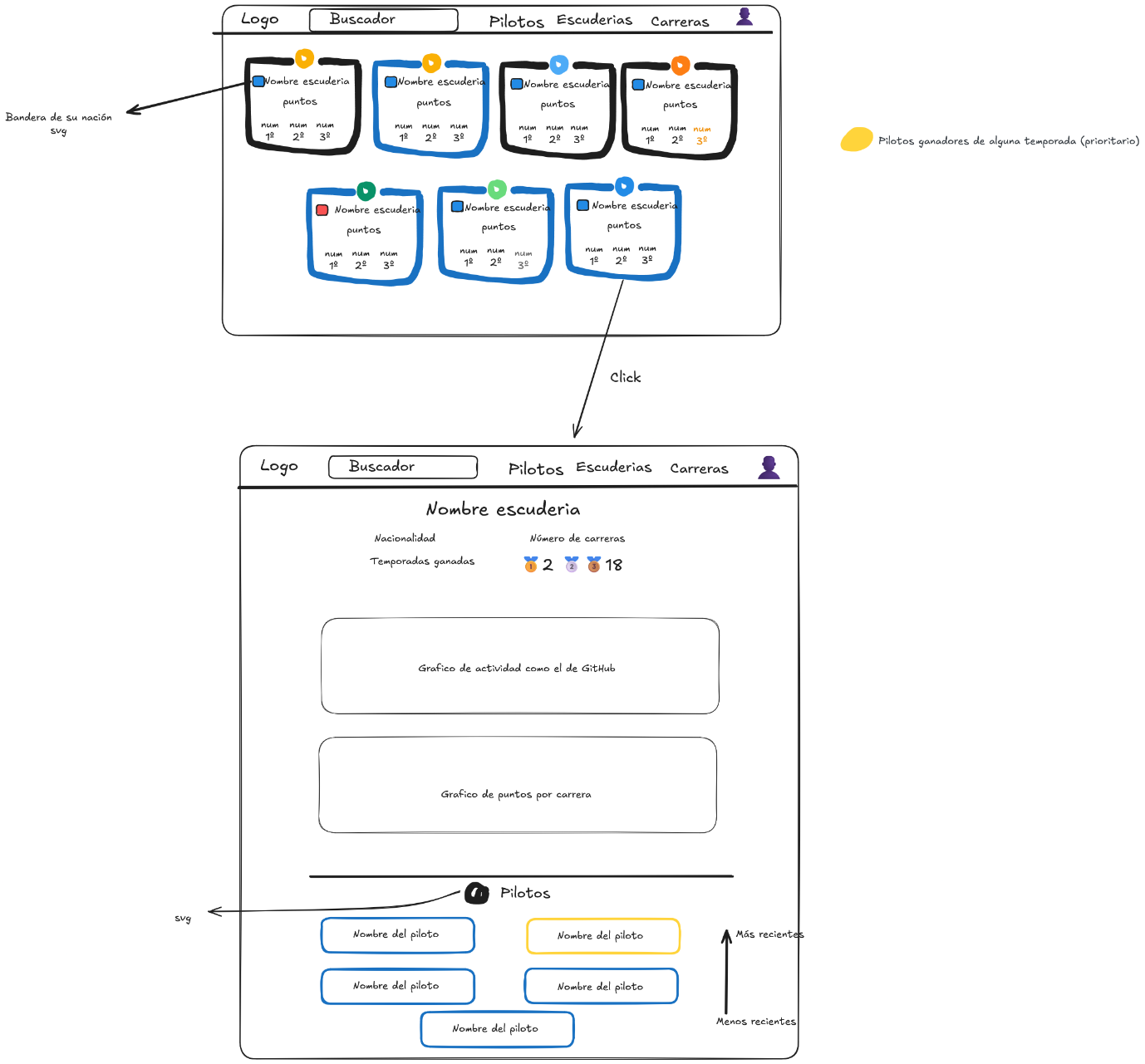


Figura 3.2. Boceto de la página de escuderías y su perfil

### Prototipos finales



Figura 3.3. Boceto del funcionamiento del menú de navegación

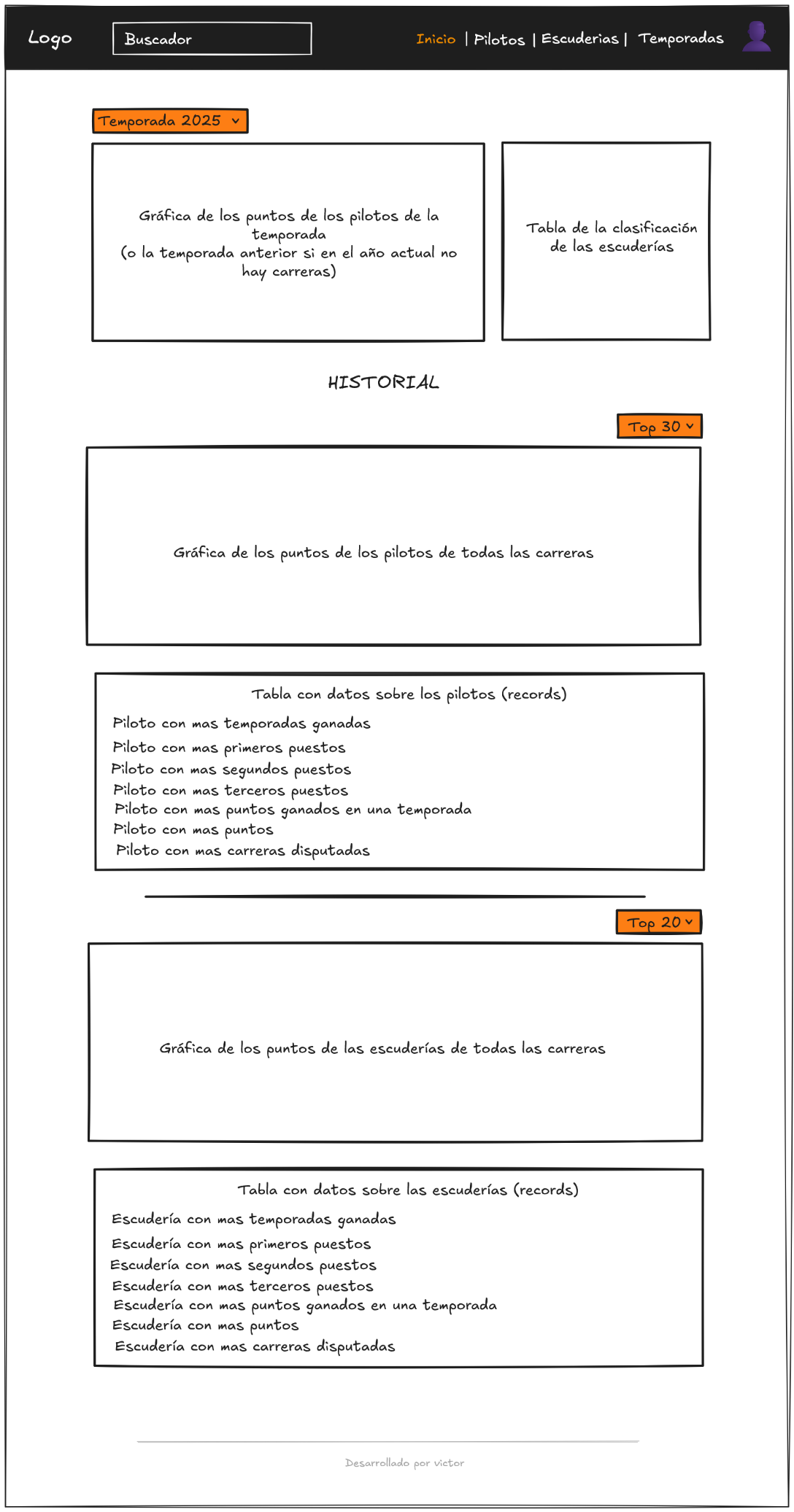


Figura 3.4. Boceto de la página de inicio

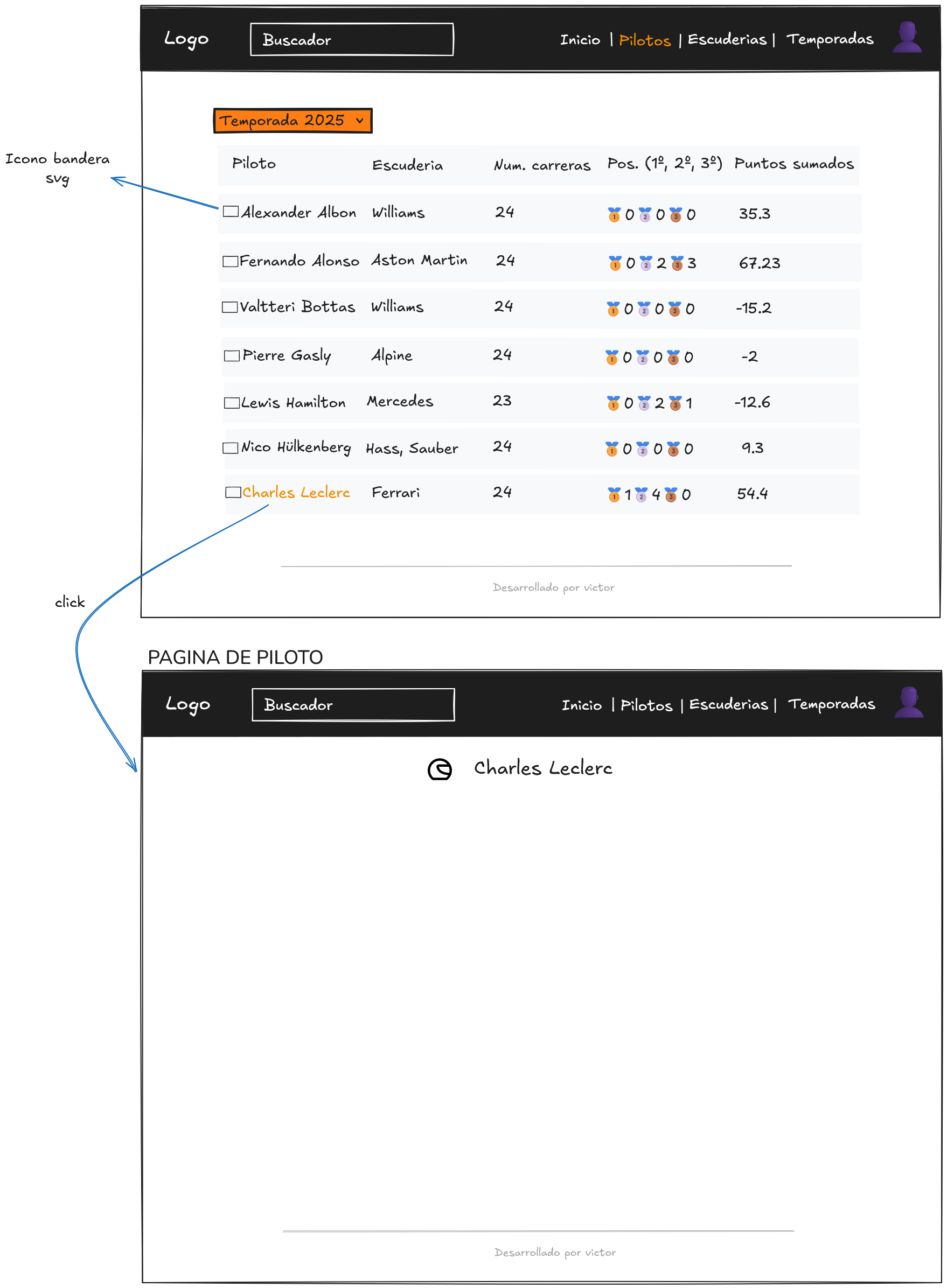


Figura 3.5. Boceto de la página de pilotos

## Diagrama relacional

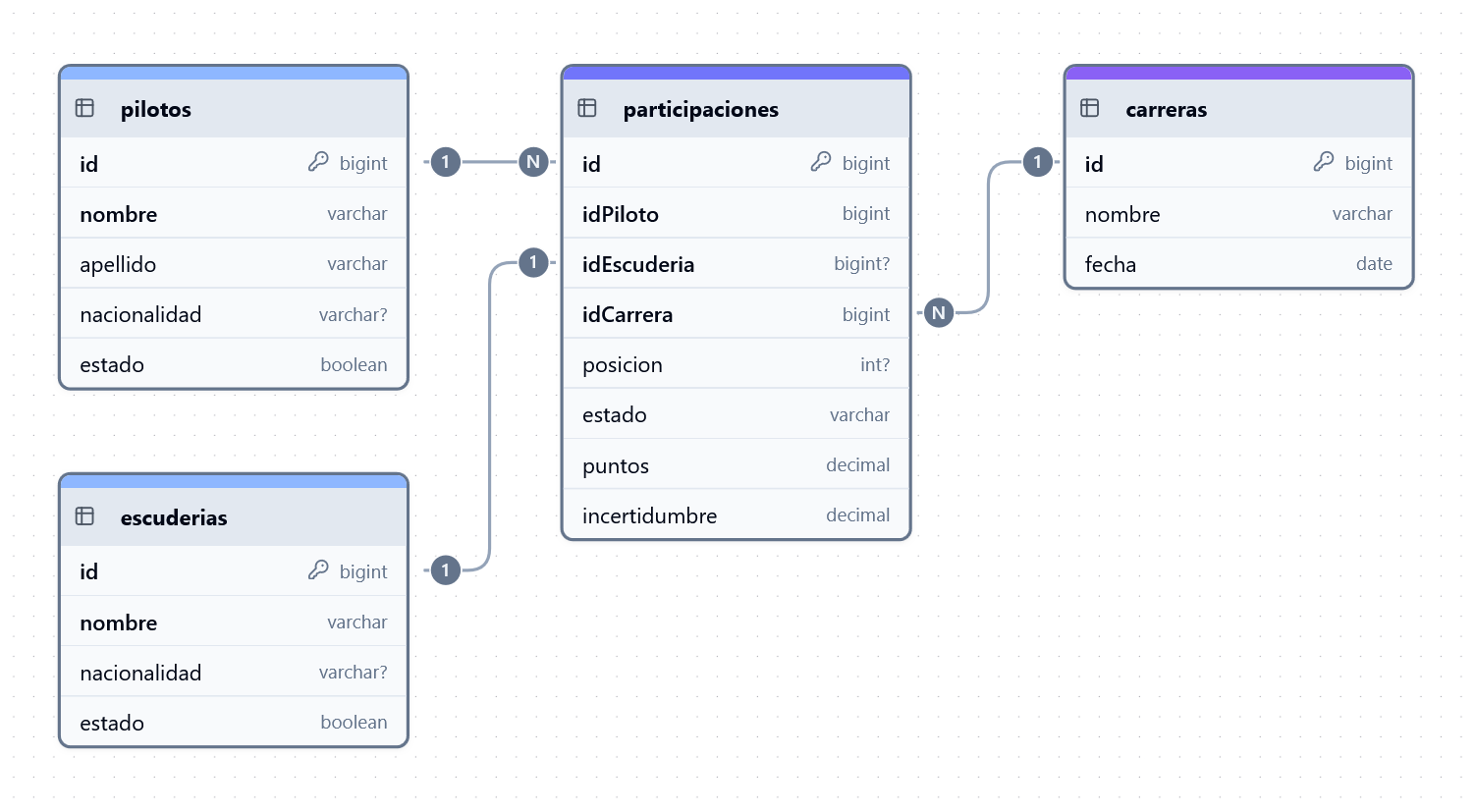


Figura 3.6. Diagrama relacional de la base de datos

## Paleta de colores

He elegido esta paleta de colores porque transmite elegancia y simplicidad, y se adapta bien tanto a navegadores con temas claros como oscuros.

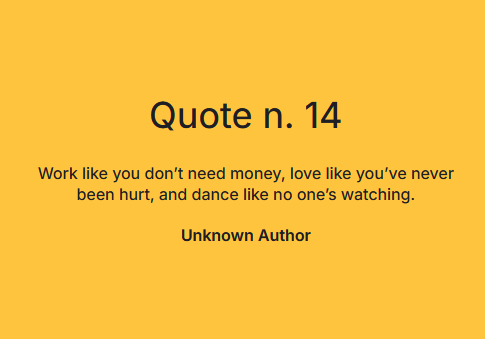
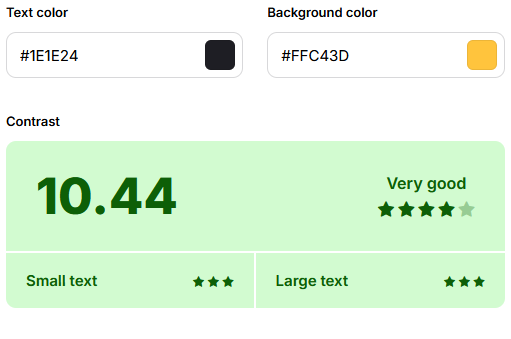


Figura 3.7. Paleta de colores

Ilustración 3.8: Contraste de la paleta de colores.

Ilustración 3.1: Contraste de la paleta de colores.

Figura 3.8. Contraste de la paleta de colores



## Tipografía

He elegido las tipografías Roboto y Open Sans porque son simples, fáciles de leer y visualmente similares entre sí. Además, opté por incluir ambas para asegurar compatibilidad en caso de que algún navegador no soporte una de ellas.

Figura 3.9. Ejemplo gráfico de las tipografías



# Pruebas

|  |  |  |
| --- | --- | --- |
| ****Nombre de la Prueba**** | ****Descripción**** | ****Fecha de Realización**** |
| Pruebas unitarias (Backend) | Verificar funciones individuales o métodos de Laravel (controladores y modelos). | Desde el inicio del desarrollo (continuamente) |
| Pruebas unitarias (Frontend) | Validar componentes de React. | Al desarrollar un nuevo componente |
| Pruebas de integración | Evaluar cómo interactúan distintos módulos entre sí (controladores y modelos). | 19/04/2025 |
| Pruebas E2E (End-to-End) | Simular el comportamiento real del usuario desde el frontend hasta la base de datos. | Cuando hay flujo funcional completo |
| Pruebas de validación | Asegurar que los formularios tengan validaciones adecuadas tanto en React como en Laravel. | 18/04/2025 |
| Pruebas de seguridad | Verificar vulnerabilidades comunes como CSRF, XSS, autenticación y autorización. | 07/05/2025 |
| Pruebas de regresión | Asegurar que nuevas funcionalidades no rompen las existentes. | Cada vez que se agrega una nueva funcionalidad (continuamente) |
| Pruebas de usabilidad | Evaluar la experiencia del usuario con la interfaz de React. | 30/04/2025 |
| Pruebas de rendimiento | Evaluar el tiempo de respuesta del backend Laravel y la carga del frontend. | 09/05/2025 |
| Pruebas de accesibilidad | Garantizar que la app pueda ser usada por personas con discapacidades. | 10/05/2025 |
| Pruebas de aceptación (UAT) | Probar que la app cumple con los requisitos. | 11/05/2025 |

Tabla 4.1: Pruebas realizadas.

# Diario de trabajo

**09/10/2024 - 12/01/2025**

* Desarrollando varias propuestas para el proyecto, incluyendo descripción de las funcionalidades y resumen de las tablas, atributos y relaciones de la base de datos.

**13/01/2025**

* Seleccionando definitivamente el tema del proyecto.

**14/01/2025**

* Investigando las tecnologías a emplear en el desarrollo.

**18/01/2025**

* Diseñando la portada y definiendo los puntos principales de la memoria; continuando la investigación tecnológica.

**19/01/2025 - 22/01/2025**

* Diseñando el esquema inicial de la base de datos.

**23/01/2025**

* Corrigiendo el diseño de la base de datos con la profesora del módulo de Desarrollo Web en Entorno Servidor.

**24/01/2025**

* Desarrollando el diagrama entidad-relación y el modelo relacional.

**26/01/2025**

* Investigando la fórmula para calcular la puntuación y diseñando el logotipo de la aplicación.

**27/01/2025**

* Redactando el planteamiento del problema y los objetivos; diseñando la página de gestión de pilotos.

**28/01/2025**

* Documentando el público objetivo del proyecto.

**05/02/2025**

* Analizando servidores de hosting gratuitos.

**20/02/2025**

* Rediseñando la base de datos.

**21/02/2025**

* Revisando el diseño de la base de datos con la profesora y profundizando en las tecnologías a implementar.

**22/02/2025**

* Desarrollando el diagrama entidad-relación y relacional definitivo; creando la base de datos; diseñando el logo.

**23/02/2025**

* Desarrollando el backend: configurando Next.js, Prisma ORM y creando un modelo y controlador de prueba.

**24/02/2025**

* Finalizando el anteproyecto; corrigiendo la implementación de la base de datos en el backend; comprobando la operatividad del MVC.

**26/02/2025**

* Ajustando el tipo de dato de posición en la base de datos; investigando paleta de colores; documentando tecnologías a usar.

**27/02/2025**

* Programando funciones para gestionar nacionalidades de equipos y pilotos.

**28/02/2025**

* Insertando datos de prueba en la base de datos; corrigiendo errores de vistas; investigando codificación de caracteres en Prisma.

**01/03/2025**

* Implementando componente de icono bandera según nacionalidad; eliminando el campo color de la tabla de equipos de la base de datos y actualizando diagramas.

**02/03/2025**

* Investigando el uso de APIs en la aplicación.

**03/03/2025**

* Diseñando el diagrama de Gantt del proyecto.

**04/03/2025**

* Desarrollando la página de gestión de pilotos: tabla, botones de borrar y actualizar.

**05/03/2025**

* Implementando la eliminación de pilotos; creando buscador por nombre y apellido.

**06/03/2025**

* Desarrollando la creación de pilotos; diseñando la interfaz de edición.

**07/03/2025**

* Optimizando el CRUD de pilotos; retocando el logo de la aplicación.

**08/03/2025**

* Abstrayendo el código del CRUD de pilotos en componentes reutilizables.

**09/03/2025**

* Documentando la planificación en la memoria; continuando la abstracción del CRUD.

**10/03/2025**

* Desarrollando la página de gestión de escuderías.

**11/03/2025**

* Desarrollando la página de gestión de carreras.

**12/03/2025**

* Implementando funcionalidades en los controladores.

**13/03/2025**

* Optimizando funciones de controladores; reestructurando la gestión de carreras; redactando propuestas de mejora.

**14/03/2025**

* Abstrayendo código de gestión de pilotos; optimizando el componente de bandera.

**15/03/2025**

* Creando componentes para modales, formularios y tablas de gestión de pilotos.

**16/03/2025**

* Desarrollando hook para gestión de pilotos; implementando middleware de validación; ampliando la API de pilotos; realizando pruebas de codificación.

**17/03/2025**

* Corrigiendo respuestas y optimizando código de la API; intentando mensajes de alerta tras peticiones; finalizando abstracción de la gestión de pilotos.

**18/03/2025 - 19/03/2025**

* Diseñando y refinando el logotipo de la aplicación.

**20/03/2025**

* Investigando Supabase y configurando base de datos en la plataforma.

**21/03/2025**

* Importando esquema de Supabase a Prisma.

**22/03/2025**

* Mejorando menú de navegación; implementando metadatos y componente “<head>” personalizable; añadiendo favicon; creando página de error 404.

**23/03/2025**

* Integrando API de Supabase y configurando políticas de select; probando peticiones HTTP con RapidAPI.

**24/03/2025 - 25/03/2025**

* Insertando y actualizando datos mediante RapidAPI con autenticación de usuario.

**26/03/2025**

* Reiniciando proyecto con Laravel + React; probando conexión de la base de datos con Turso.

**27/03/2025**

* Instalando Laravel + React; probando componentes en el dashboard.

**28/03/2025**

* Editando diseño del header; modificando paleta de colores.

**29/03/2025**

* Desarrollando el header; definiendo la página de inicio.

**30/03/2025**

* Creando migraciones de la base de datos; estableciendo rutas; comenzando la vista index de pilotos.

**31/03/2025**

* Configurando “fillable” y atributos del modelo Pilotos.

**01/04/2025**

* Implementando columnas y celdas en la tabla de pilotos.

**02/04/2025**

* Desarrollando la tabla de pilotos: columnas visibles e invisibles dependiendo del tipo de usuario; desplegable por temporada; responsividad.

**03/04/2025**

* Ajustando redirección al cerrar sesión; abstrayendo código de la tabla de pilotos.

**05/04/2025**

* Configurando visibilidad de columnas según temporada; desarrollando botón de creación y estructura de la página de pilotos.

**06/04/2025**

* Optimizando rutas a métodos del controlador; explorando formulario de actualización.

**07/04/2025**

* Desarrollando formularios de creación y edición; corrigiendo estilos de bordes.

**10/04/2025**

* Completando CRUD de pilotos.

**11/04/2025**

* Desarrollando CRUD de escuderías.

**12/04/2025**

* Diseñando vista y perfil de carreras.

**13/04/2025**

* Implementando drag-and-drop en formulario de clasificación de carrera con dnd-kit.

**14/04/2025**

* Documentando el apartado de la paleta de colores de la aplicación.

**15/04/2025**

* Configurando tipografía; ajustando logo en el header.

**16/04/2025**

* Añadiendo atributo de estado a tablas de pilotos y escuderías; ajustando CRUD para el nuevo campo.

**17/04/2025**

* Desarrollando formulario y validación de clasificación de carrera.

**18/04/2025**

* Implementando sistema de puntuación de pilotos al crear carreras.

**19/04/2025**

* Añadiendo funciones de actualizar y borrar carreras; creando datos para la página de carreras.

**20/04/2025**

* Diseñando icono con fondo para la documentación; creando datos para la página de pilotos.

**21/04/2025**

* Corrigiendo datos de la página de pilotos.

**22/04/2025**

* Añadiendo datos en la página de escuderías; corrigiendo estilos de tablas e incluyendo todos los tamaños de fuente en la URL de importación de las tipografías.

**23/04/2025**

* Añadiendo columna de “mejor piloto” en la página de carreras; optimizando consultas.

**24/04/2025**

* Implementando perfil de pilotos con gráficos.

**25/04/2025**

* Probando despliegues en Laravel Cloud, Railway y Render; optimizando modelo Pilotos.

**26/04/2025**

* Elaborando README y documentación de despliegue local y en AWS.

**27/04/2025**

* Corrigiendo fórmula de cálculo de puntos en participaciones.

**28/04/2025**

* Diseñando diagrama de aplicación.

**30/04/2025**

* Desarrollando diagrama de casos de uso.

**01/05/2025**

* Corrigiendo redireccionamientos en controladores; eliminando opción de registro; ajustando datos enviados a perfil de pilotos; implementando perfil de escuderías; refactorizando código; corrigiendo estilos y diagramas; plasmando propuestas de ampliación en la documentación.

**02/05/2025**

* Implementando certificado SSL autofirmado; desarrollando página de temporadas.

**03/05/2025**

* Seleccionando plantilla para la presentación.

**04/05/2025**

* Creando gráficos para perfil de carreras.

**05/05/2025**

* Desarrollando y corrigiendo la mayoría de gráficas del perfil de carreras; corrigiendo errores de estilo.

**06/05/2025**

* Mejorando estilos de los gráficos.

**07/05/2025**

* Optimizando perfil de carreras; añadiendo gráficos y mejorando sistema de puntuación de equipos; incorporando tablas de clasificación por temporada.

**08/05/2025**

* Corrigiendo títulos de páginas; mejorando responsividad; corrigiendo error de actualización de participaciones; refactorizando TypeScript.

**09/05/2025**

* Añadiendo metadatos a la plantilla principal.

**10/05/2025**

* Finalizando perfil de pilotos; optimizando funciones del modelo de pilotos.

**11/05/2025**

* Corrigiendo la documentación y finalizando los últimos apartados.

# Conclusiones

## Objetivos cumplidos

A lo largo del desarrollo del proyecto, se han alcanzado los principales objetivos planteados inicialmente, consolidando una base funcional sólida.

* **Sistema de gestión de carreras**: Se ha desarrollado una aplicación web funcional que permite gestionar competiciones de manera eficiente. Aunque el sistema cumple su propósito, se han identificado una posible mejora en la eficiencia de carga de páginas, aspecto que se contempla para futuras optimizaciones.
* **Sistema de puntuación**: El proyecto integra con éxito un modelo de puntuación inspirado en TrueSkill, no obstante, se reconocen áreas susceptibles de mejora que requieren pruebas adicionales para afinar el modelo y validar su comportamiento en distintos escenarios.
* **Optimización de la eficiencia y experiencia de usuario**: La aplicación ofrece una experiencia de usuario agradable, con una interfaz intuitiva y navegación fluida. Sin embargo, aún existen aspectos que podrían mejorarse, especialmente en términos de tiempos de carga y refinamiento de la interacción.

## Propuestas de mejora y ampliaciones futuras

* Incorporar nuevos idiomas además del inglés, incluyendo español, portugués, francés e italiano. Esto permitirá mejorar la experiencia de los usuarios al ofrecerles una interfaz en su idioma. La implementación incluirá la traducción de la interfaz y un sistema de selección de idioma en la configuración, asegurando una navegación fluida y adaptada a cada usuario.
* Automatizar completamente el proceso de creación y configuración de la aplicación para facilitar su despliegue por terceros, mediante scripts, permitiendo que cualquier usuario externo pueda clonar el proyecto, configurar su entorno y desplegar la aplicación de forma rápida, estandarizada y sin necesidad de conocimientos técnicos avanzados.
* Incorporar en la página de inicio una funcionalidad que permita comparar el rendimiento de hasta cuatro pilotos o escuderías a lo largo del tiempo, mediante gráficos. Esta funcionalidad facilitará a los usuarios el análisis de estadísticas y la comparación de desempeños en distintas temporadas, mejorando la experiencia de navegación.

# Guía

## Despliegue

### Requisitos previos

* PHP versión 8.2 o superior
* Node.js versión 22.14 o superior
* Composer versión 2.8 o superior

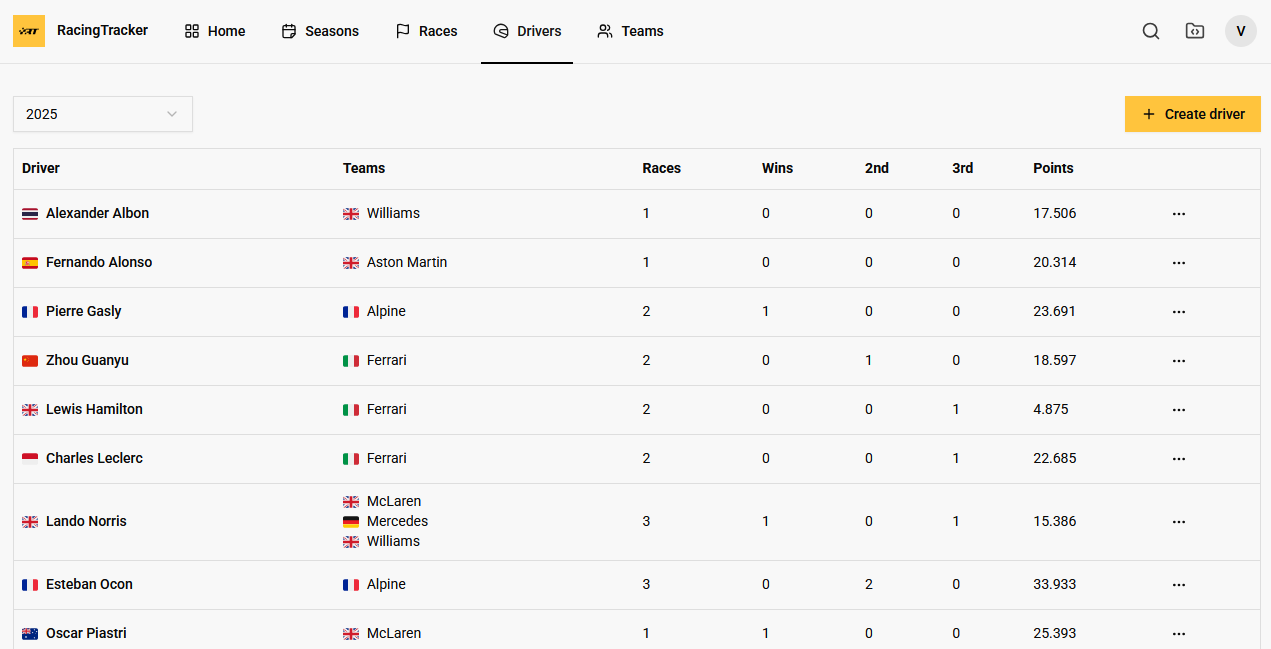
### Instalación

1. Clona el repositorio: *git clone* [*https://github.com/Qv1ko/RacingTracker.git*](https://github.com/Qv1ko/RacingTracker.git)
2. Cambia al directorio del proyecto: *cd RacingTracker*
3. Instala las dependencias:*composer install && npm install*
4. Copia el archivo *.env.example* y renómbralo a *.env*: *cp .env.example .env*
5. Genera la clave de la aplicación: *php artisan key:generate*
6. Ejecuta las migraciones de base de datos: *php artisan migrate*
7. Ejecuta los seeders (datos de prueba): *php artisan db:seed*
8. Inicia el servidor: *npm run build && php artisan serve*
9. Abre tu navegador y visita: *http://localhost:8000*

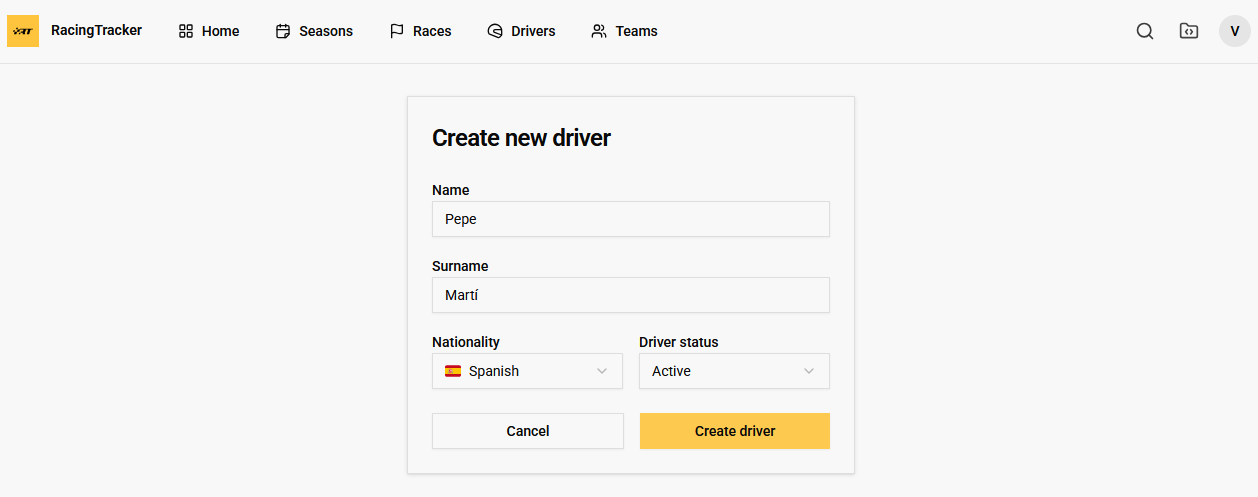
## Guía de administración

### Crear pilotos

1. Desde la página “Drivers” pulsa el botón “Create driver” para abrir el formulario.

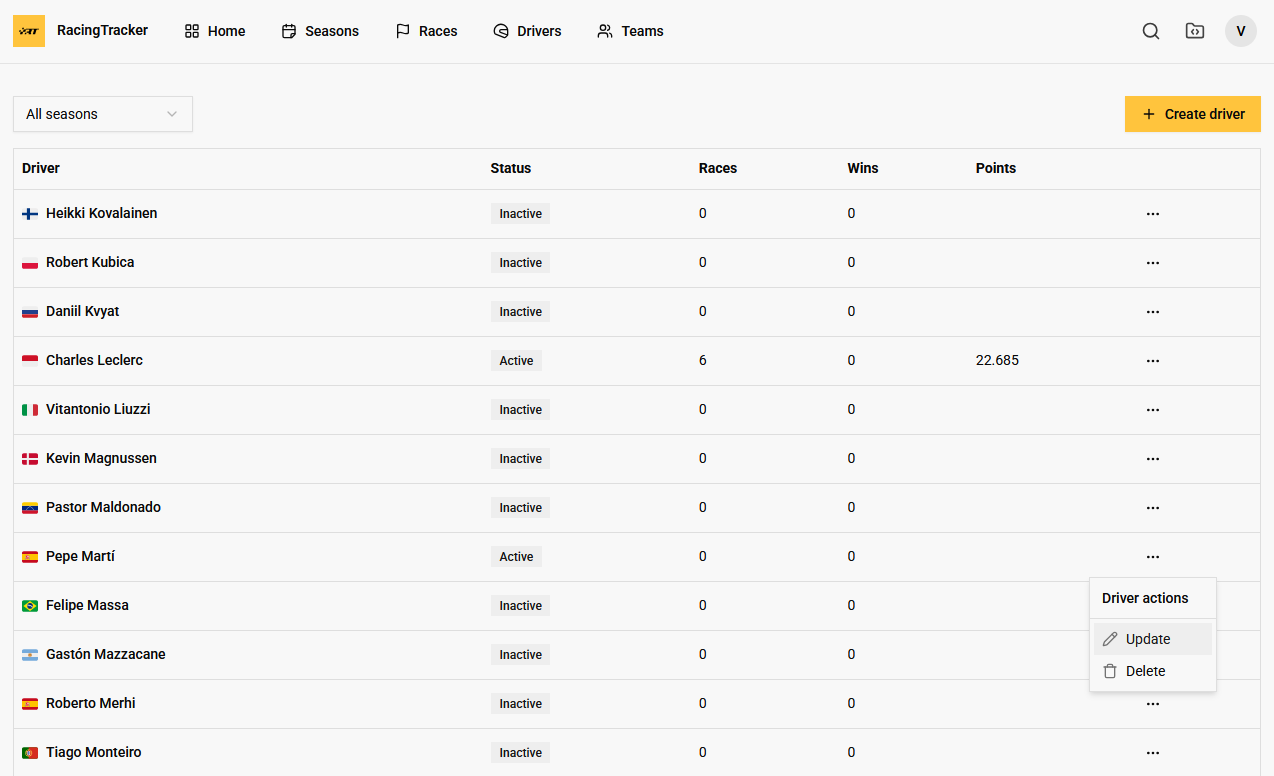


1. Rellena los campos del formulario y pulsa “Create driver”.

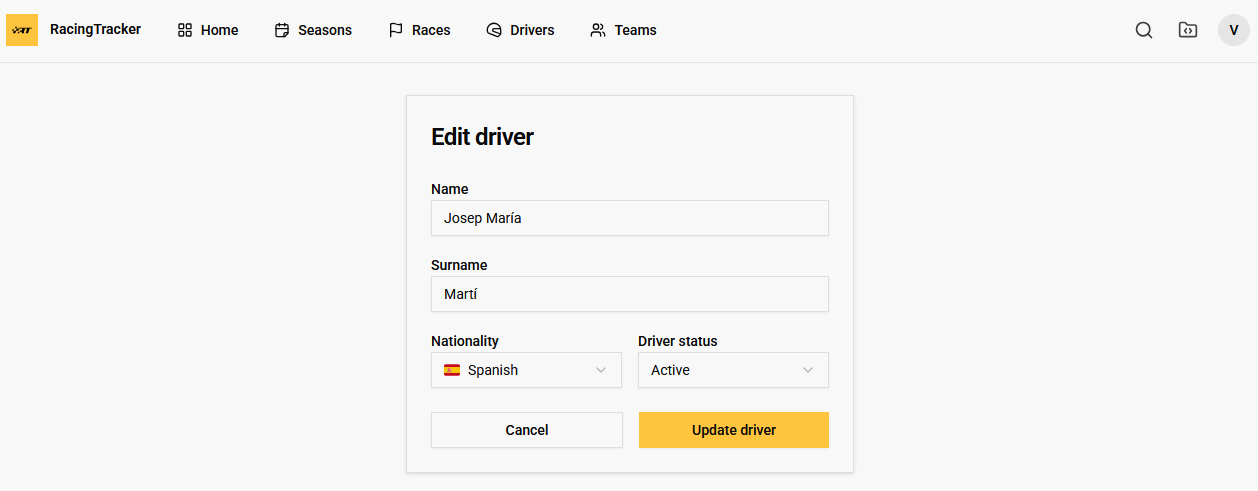


### Editar pilotos

1. Desde la página de “Drivers”, pulsa los tres puntos de la derecha del piloto que quieras editar, en el menú emergente pulsa “Update” para entrar en el formulario.

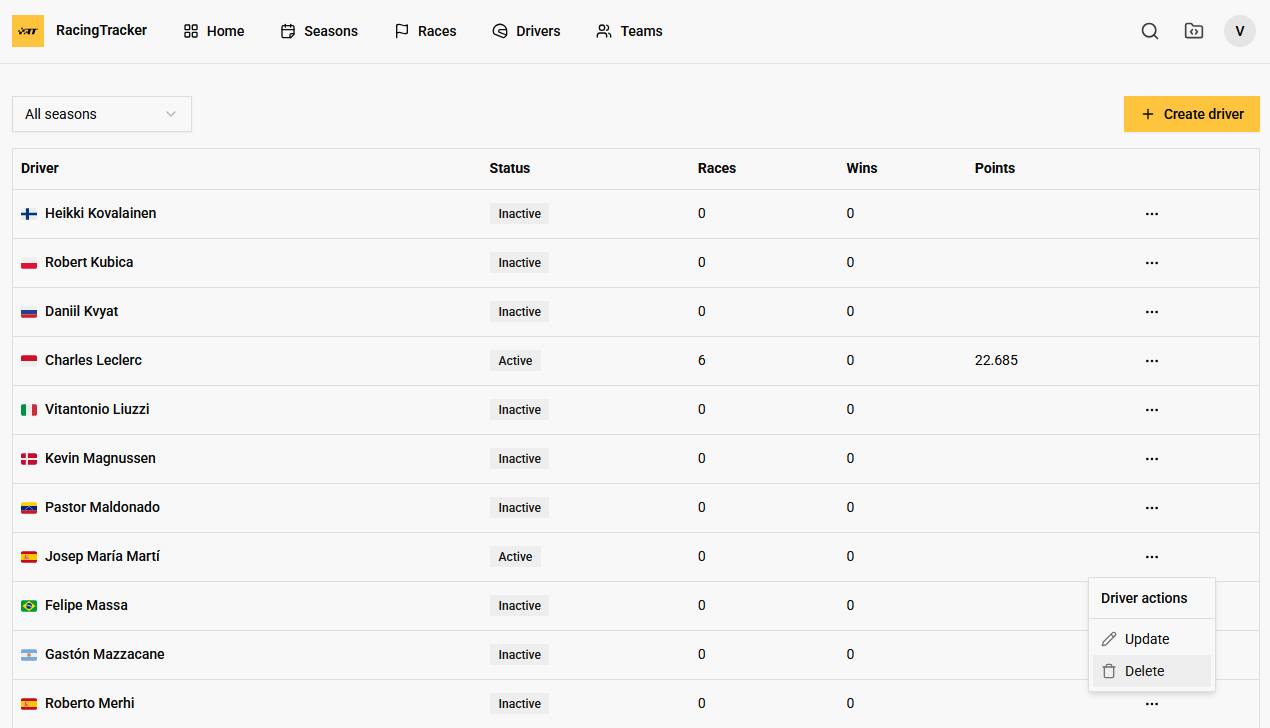


1. Realiza los cambios y pulsa el botón “Update driver”.

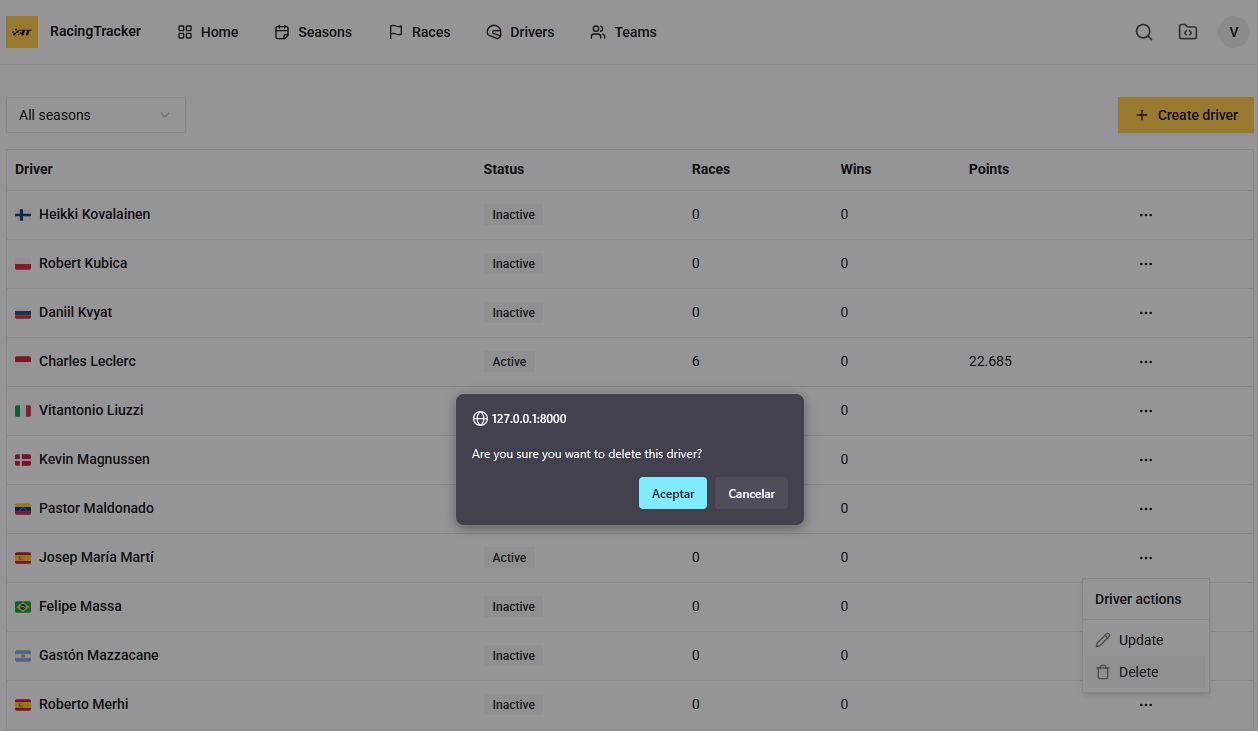


### Borrar pilotos

1. Desde la página de “Drivers”, pulsa los tres puntos de la derecha del piloto que quieras eliminar, en el menú emergente pulsa “Delete”.

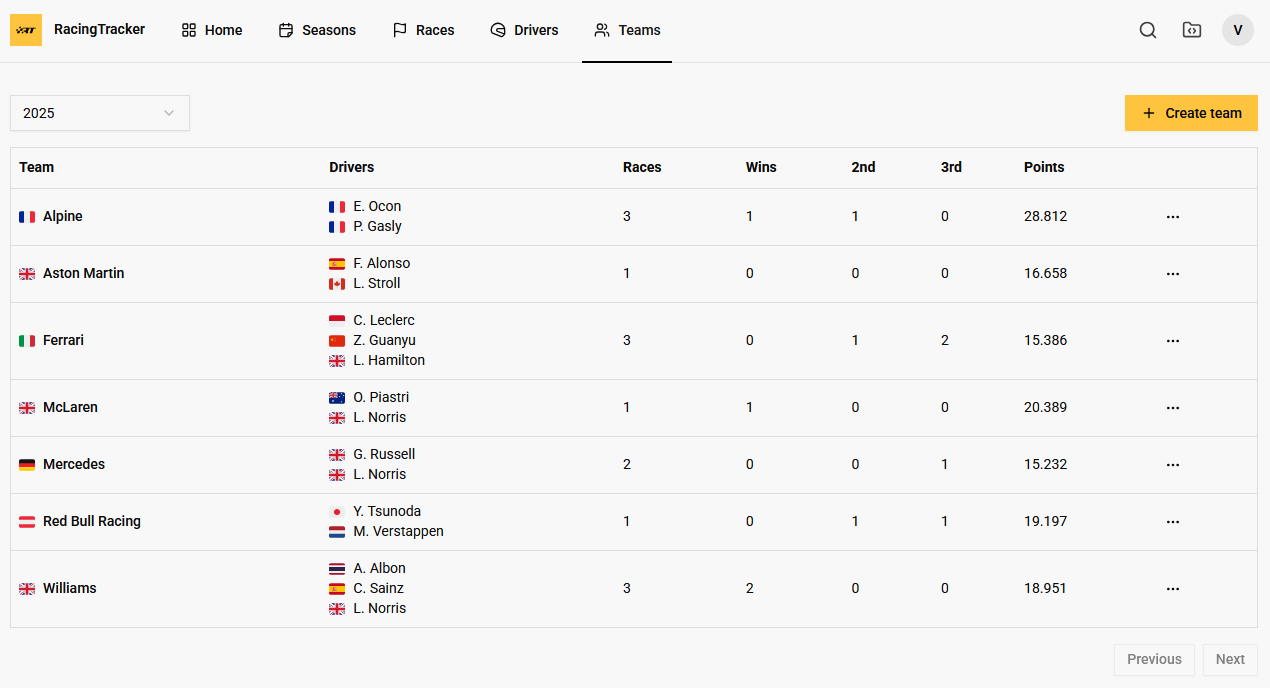


1. Desde la ventana emergente pulsa “Aceptar” para borrar el piloto.

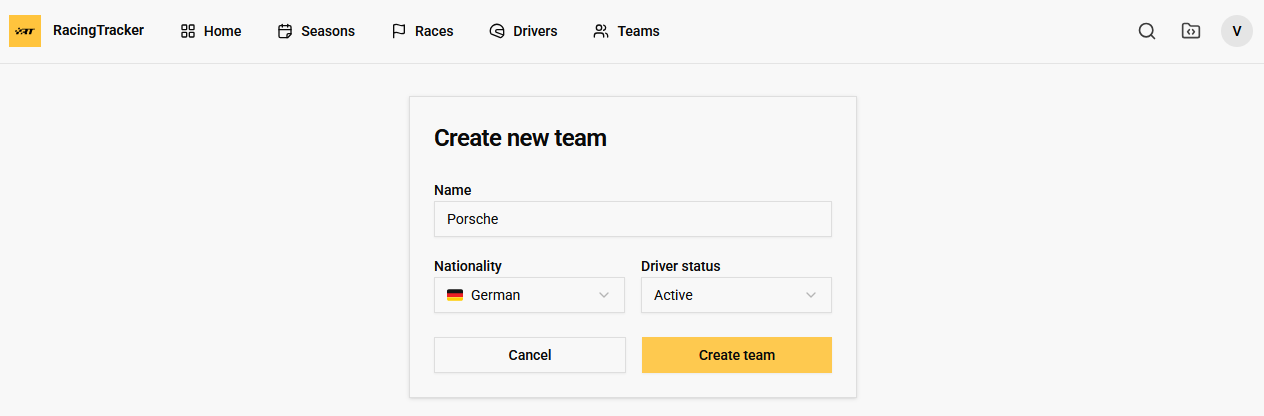


### Crear escuderías

1. Desde la página “Teams” pulsa el botón “Create team” para abrir el formulario.

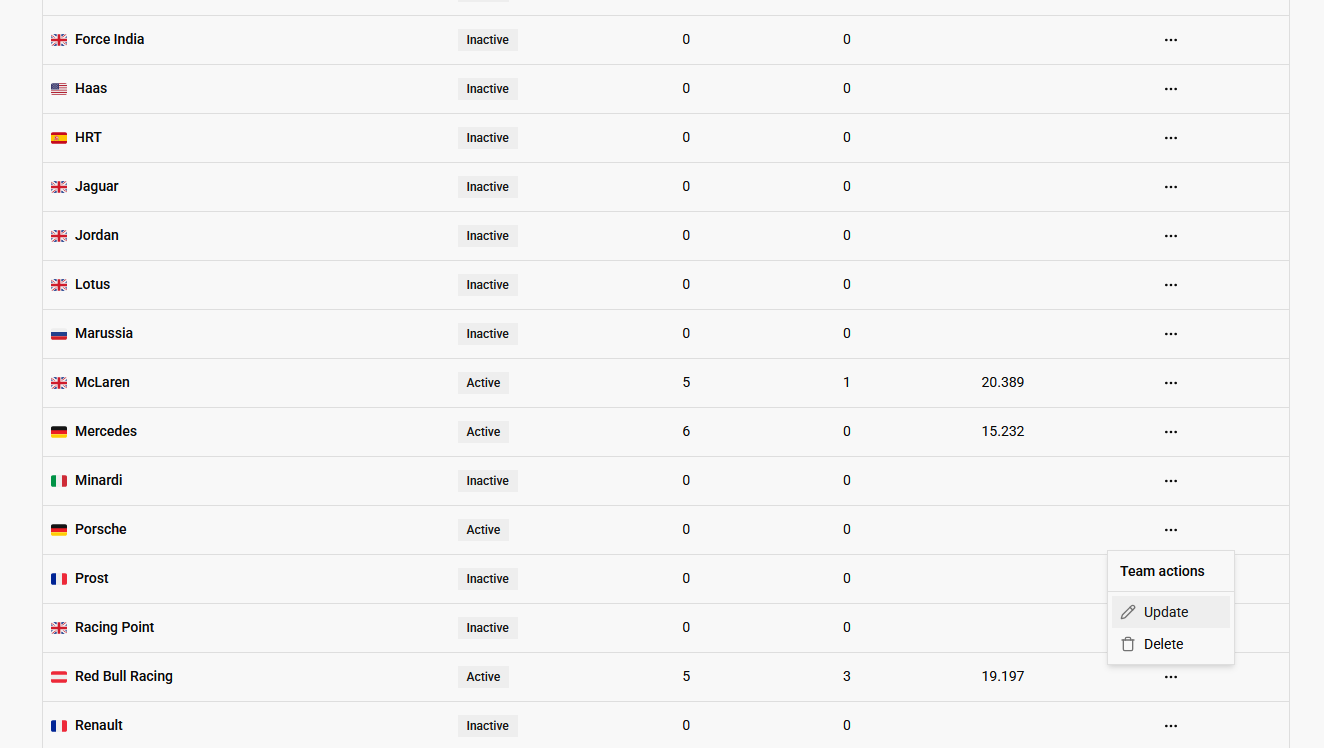


1. Rellena los campos del formulario y pulsa “Create team”.

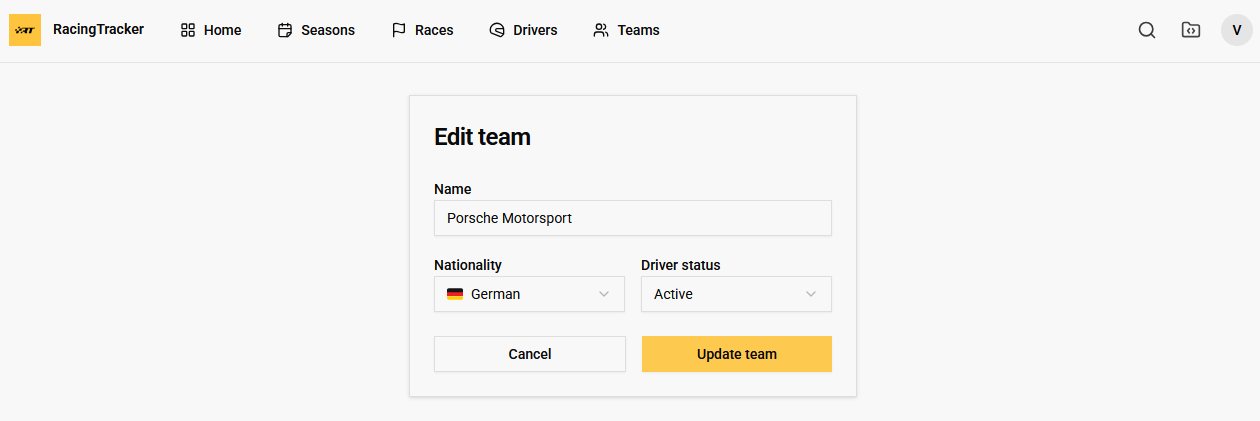


### Editar escuderías

1. Desde la página de “Teams”, pulsa los tres puntos de la derecha de la escudería que quieras editar, en el menú emergente pulsa “Update” para entrar en el formulario.

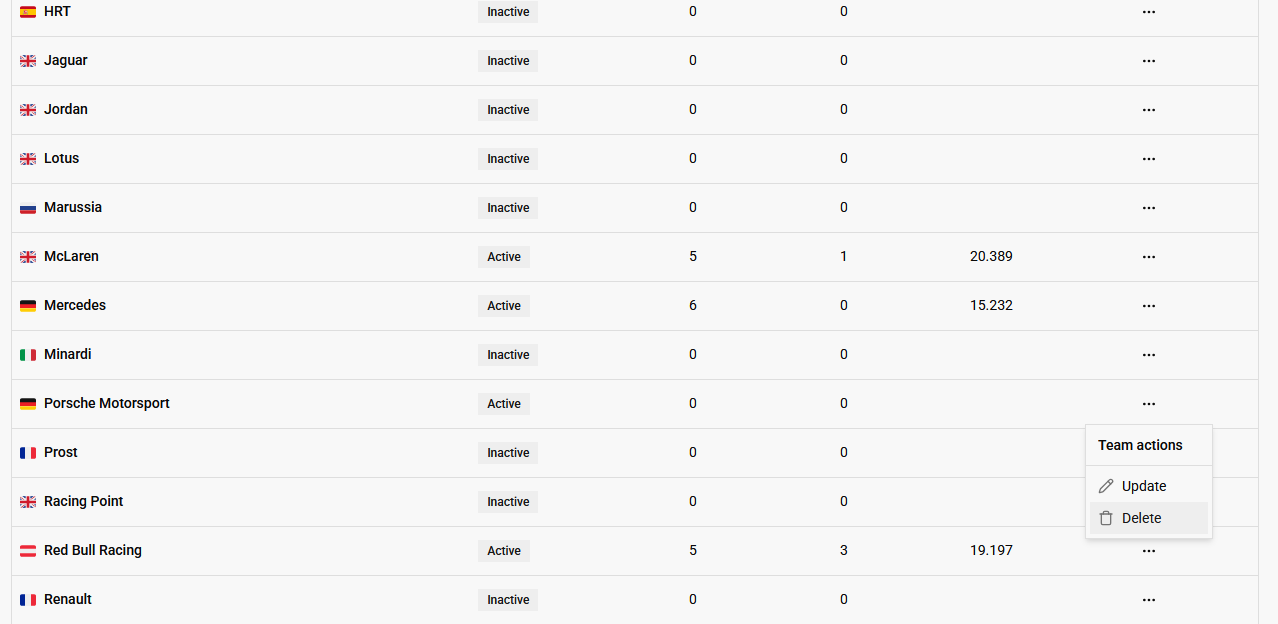


1. Realiza los cambios y pulsa el botón “Update team”.

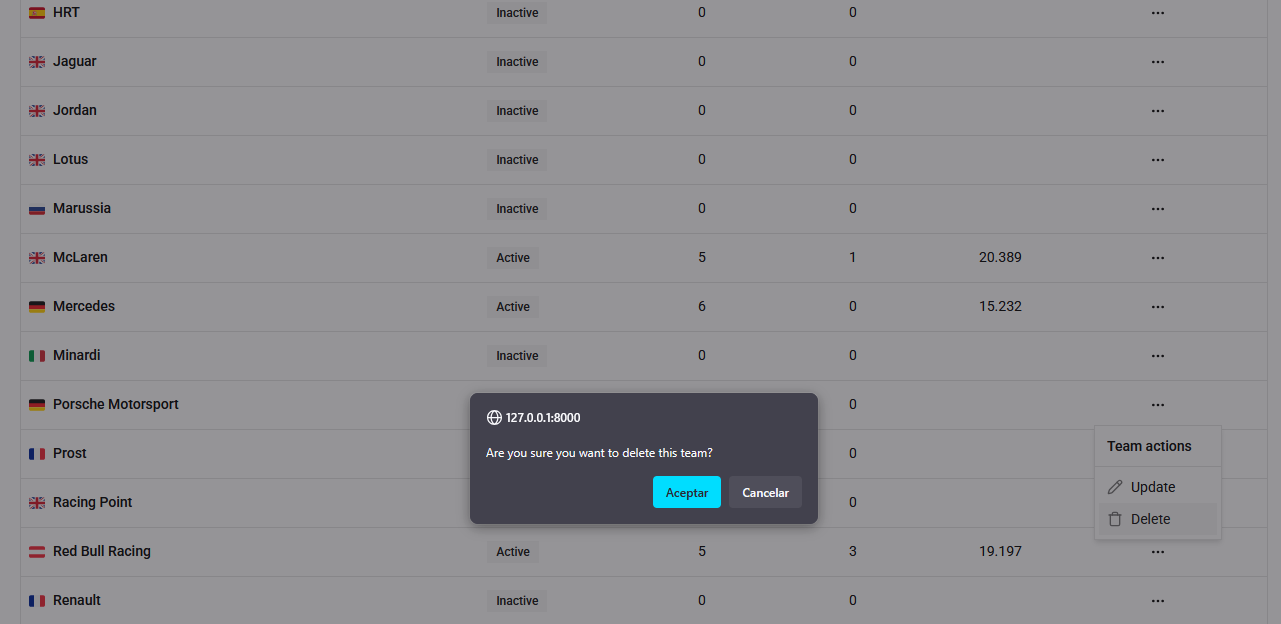


### Borrar escuderías

1. Desde la página de “Teams”, pulsa los tres puntos de la derecha de la escudería que quieras eliminar, en el menú emergente pulsa “Delete”.

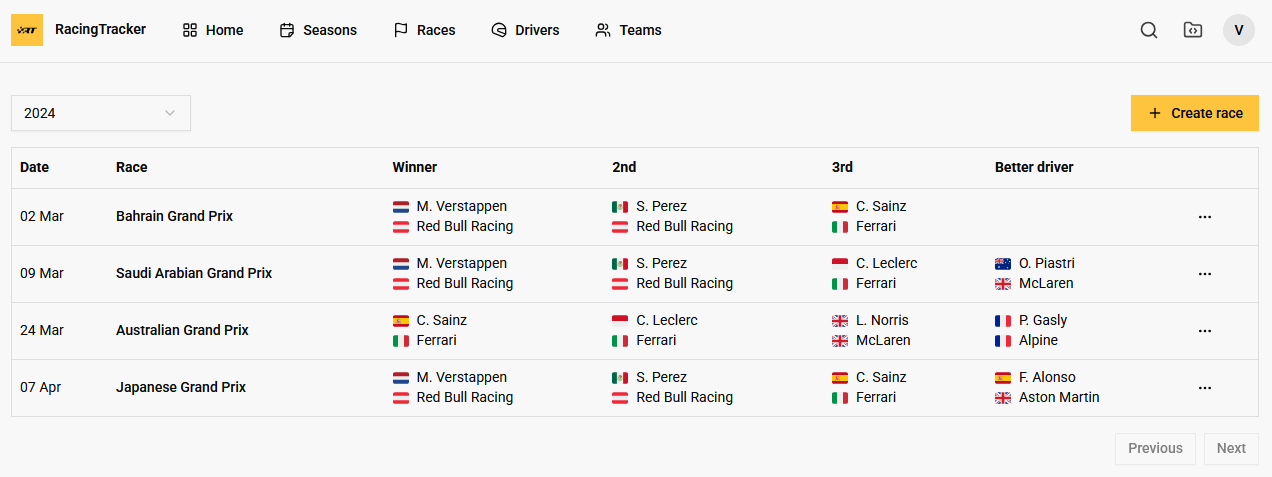


1. Desde la ventana emergente pulsa “Aceptar” para borrar la escudería.

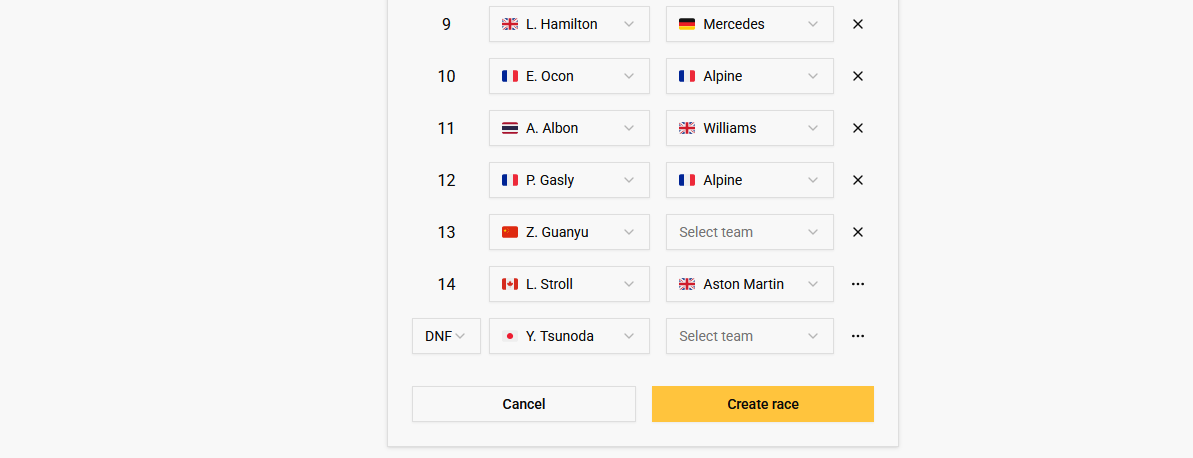
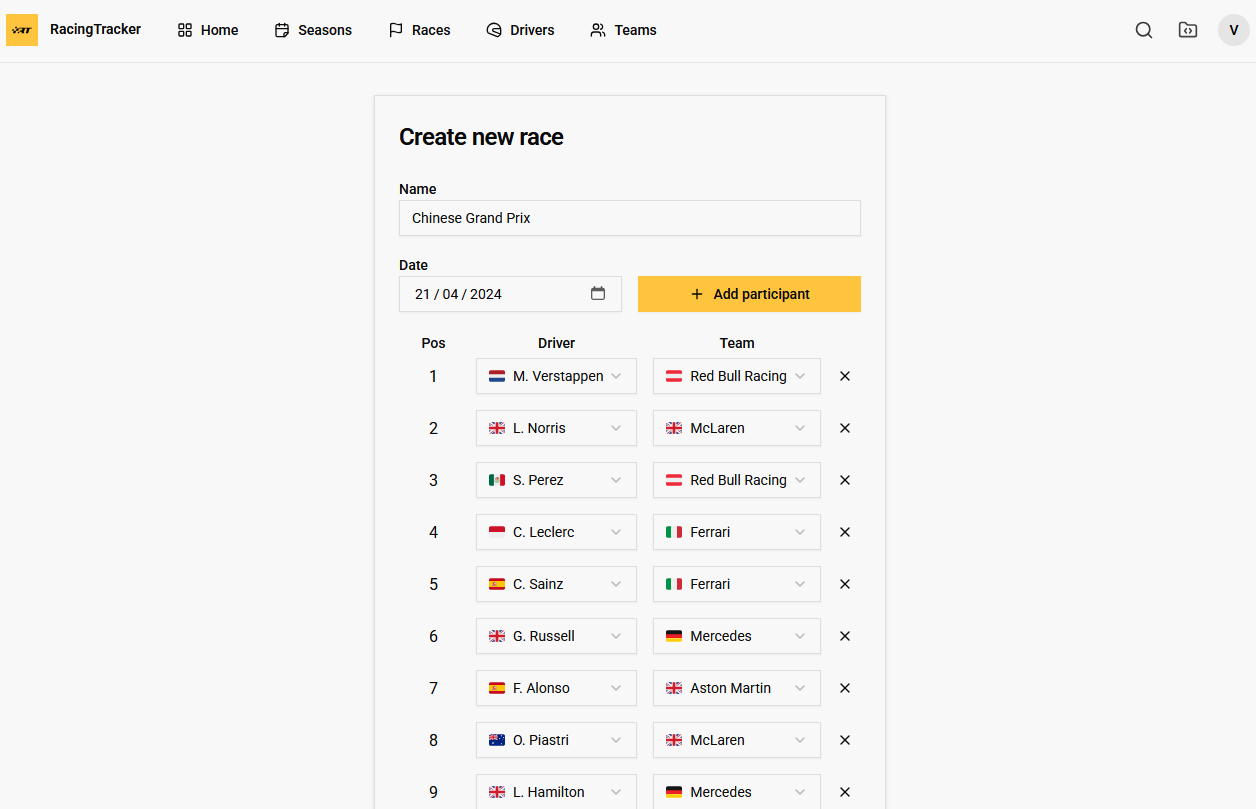


### Crear carreras

1. Desde la página “Races” pulsa el botón “Create race” para abrir el formulario.

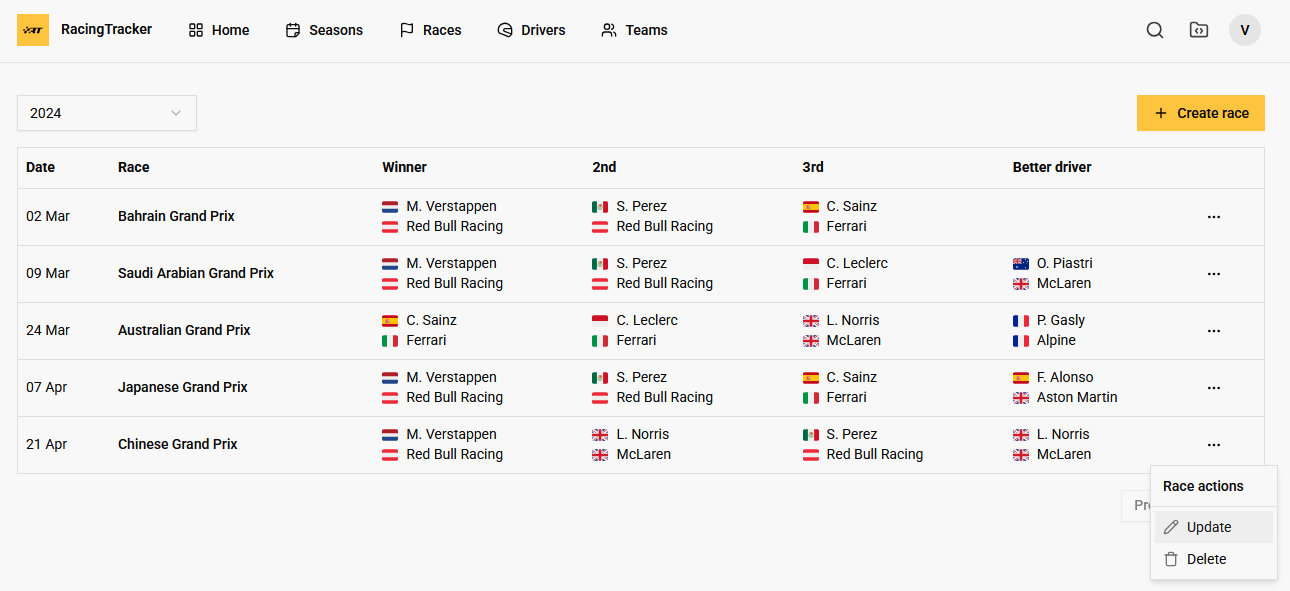


1. Rellena los datos de la carrera, pulsa el botón “Add participant” para añadir participantes de la carrera, en cada fila de un participante, puede seleccionar el piloto y la escudería. Pulsando en la X puedes eliminar al participante y pulsando en los tres puntos puedes o borrar o seleccionar una posición no numérica (DNF, NC u otras). Para guardar la carrera pulsa el botón “Create race”.

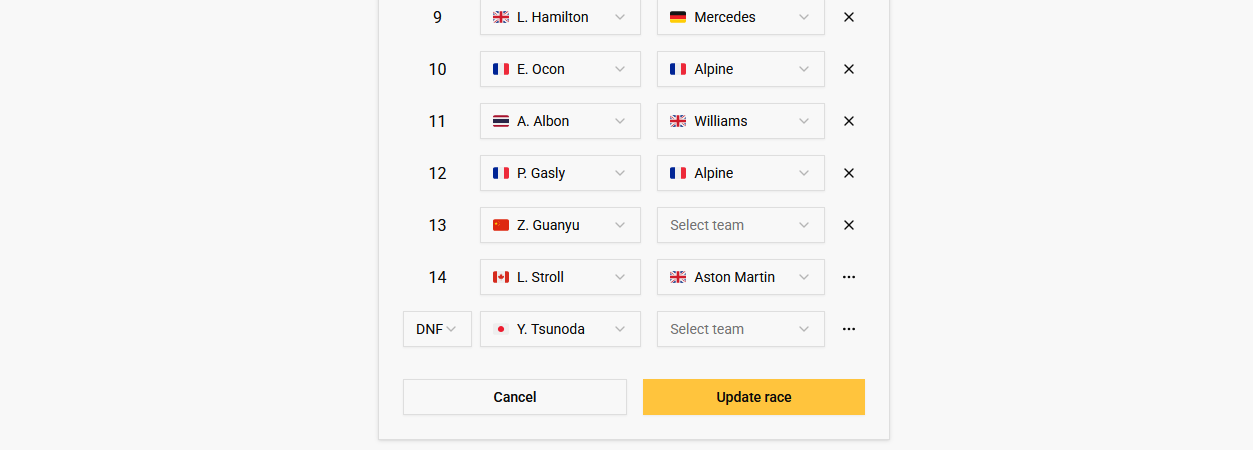


### Editar carreras

1. Desde la página de “Races”, pulsa los tres puntos de la derecha de la carrera que quieras editar, en el menú emergente pulsa “Update” para entrar en el formulario.

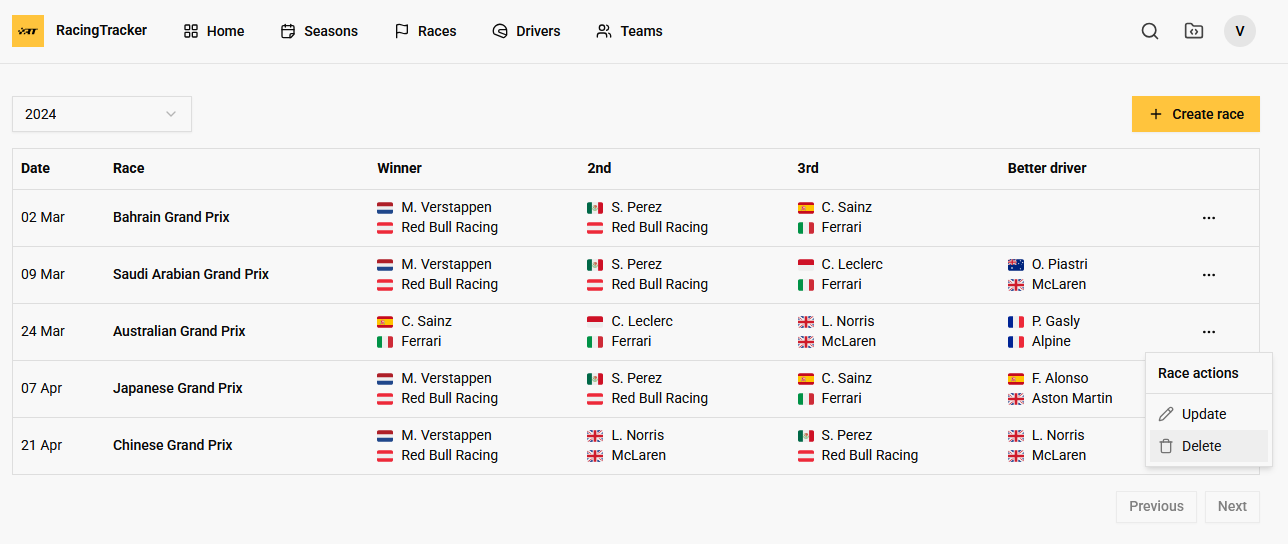


1. Realiza los cambios y pulsa el botón “Update race”.

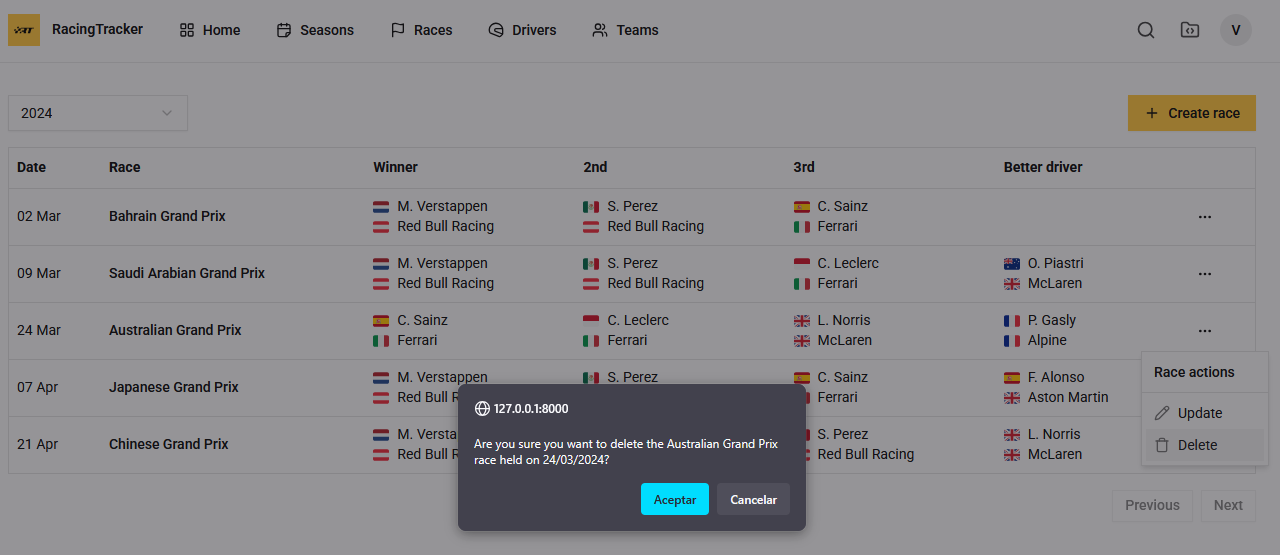


### Borrar carreras

1. Desde la página de “Races”, pulsa los tres puntos de la derecha de la carrera que quieras eliminar, en el menú emergente pulsa “Delete”.

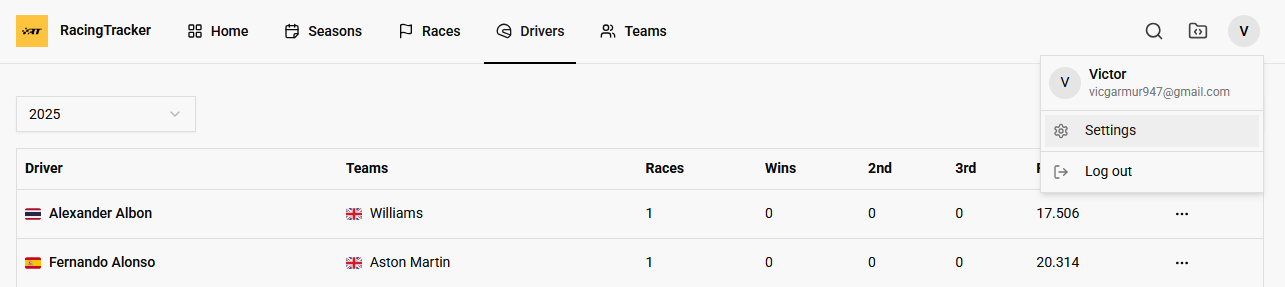


1. Desde la ventana emergente pulsa “Aceptar” para borrar la carrera.

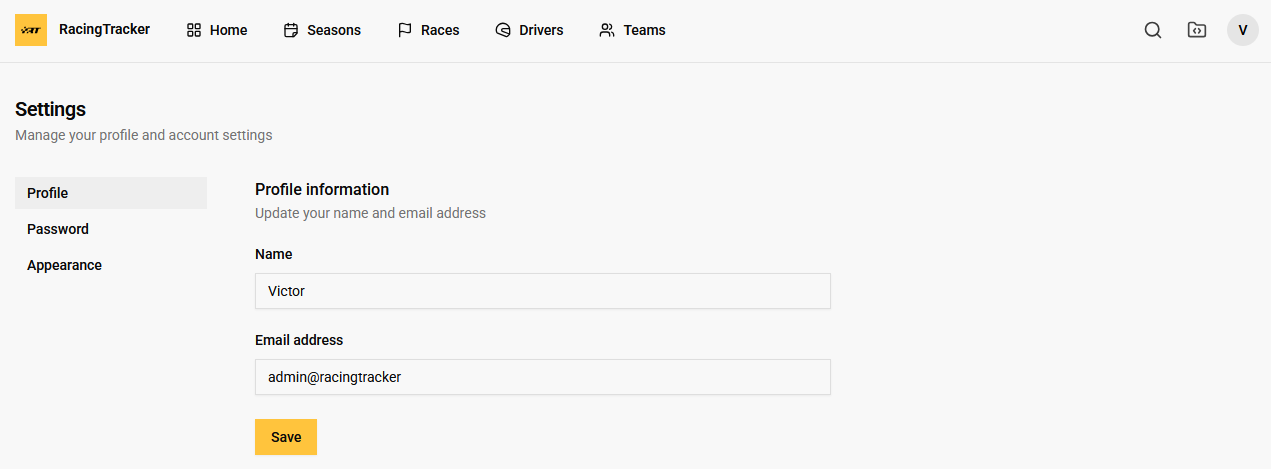


### Configuración del administrador

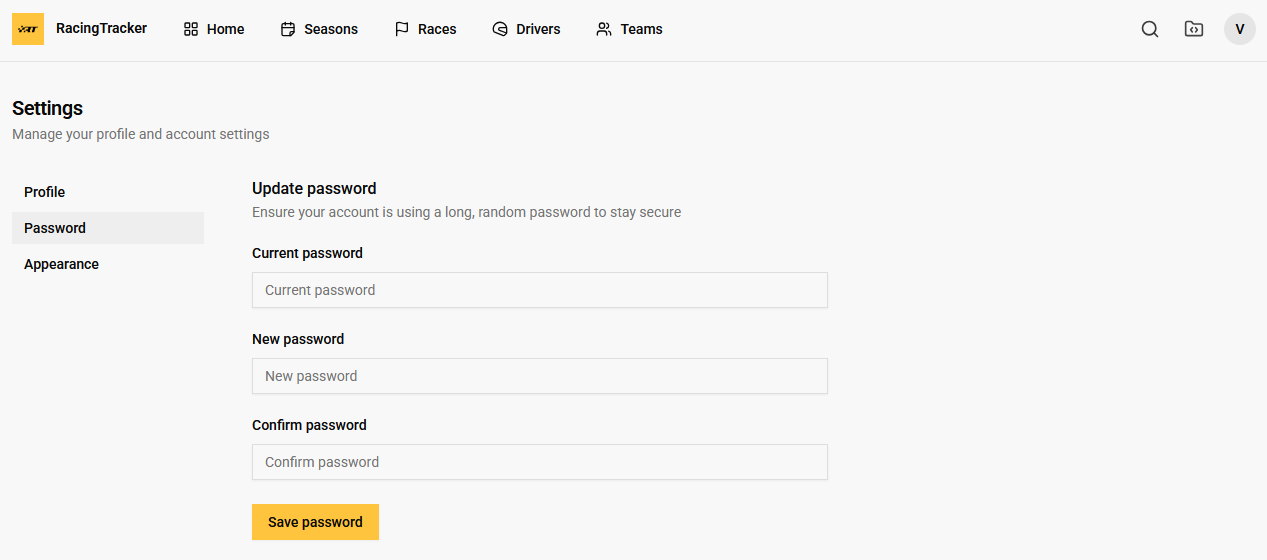
1. Pulsa el avatar de arriba a la derecha. En el menú desplegable, pulsa “Settings” para acceder a la configuración.



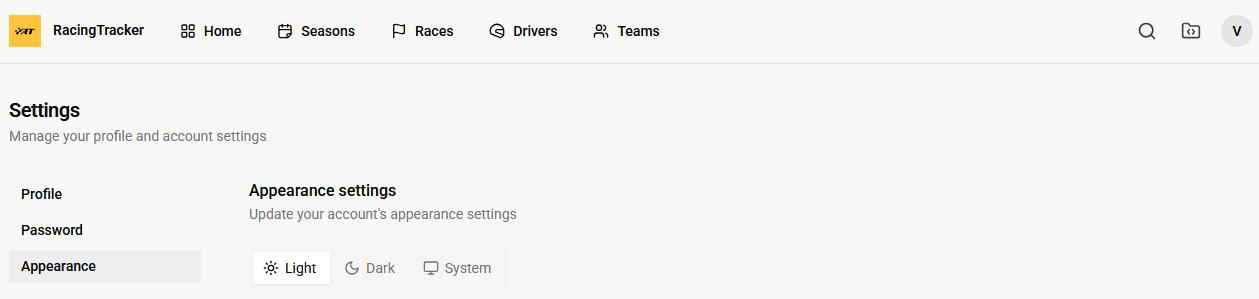
1. Desde el primer apartado, “Profile”, puedes editar el nombre del usuario y su correo. Pulsa el botón “Save”, para guardar los cambios.



1. En el segundo apartado, puedes cambiar la contraseña del usuario. Guárdala pulsando “Save password”.

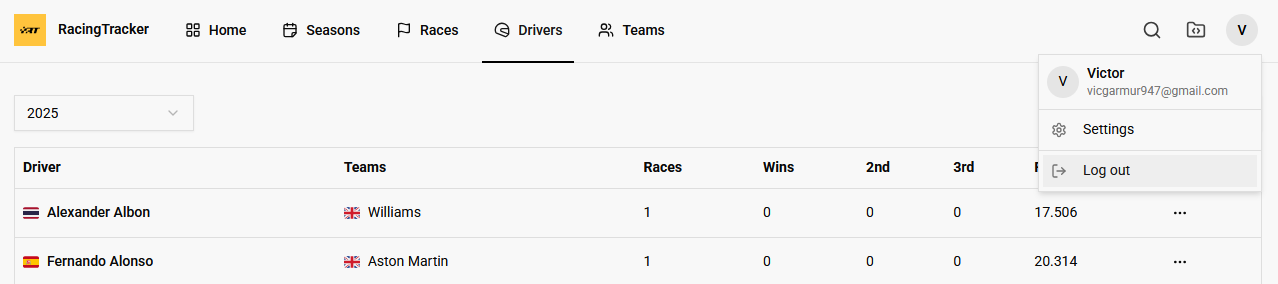


1. En el último apartado puedes cambiar el tema de la aplicación para alternar entre el modo oscuro y claro.



### Cerrar sesión

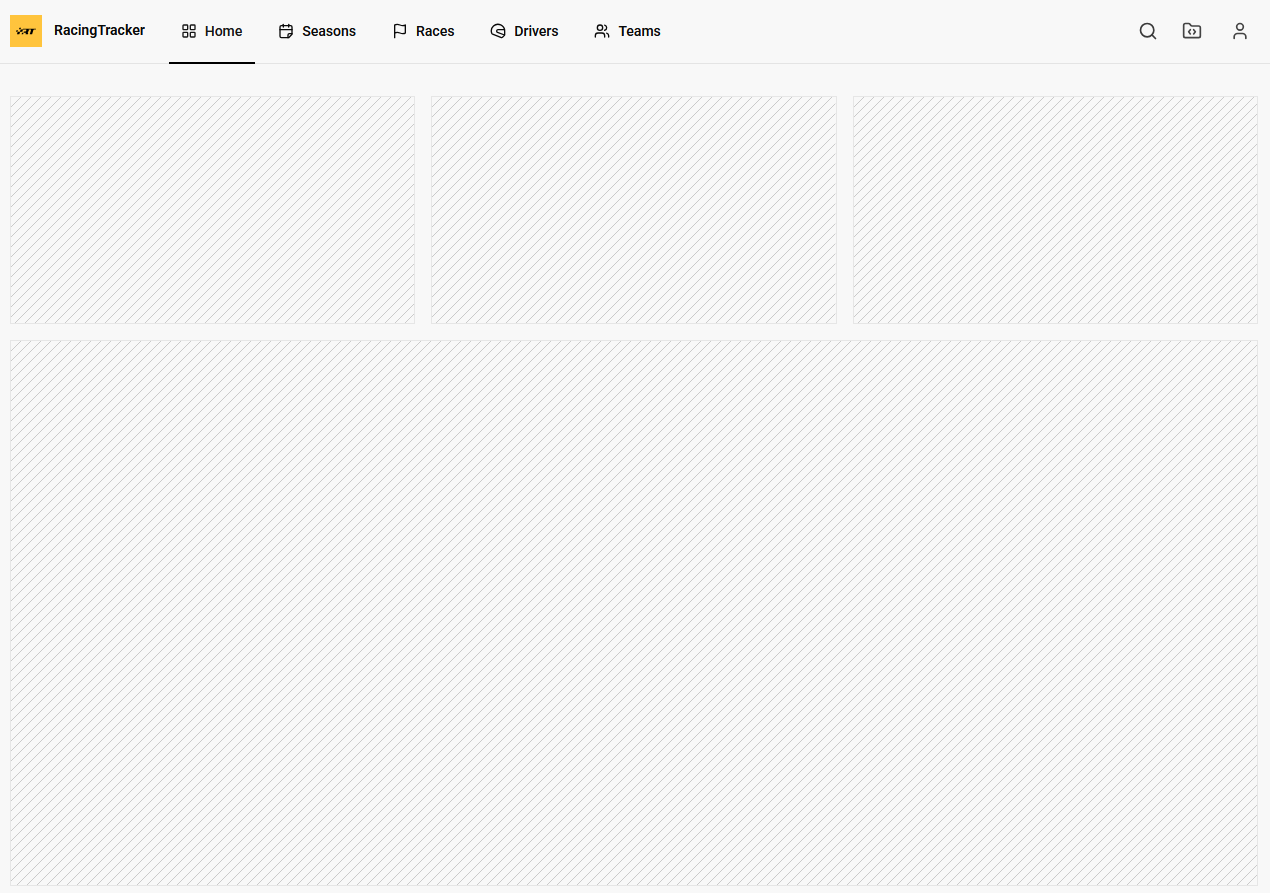
Clicando en el avatar de arriba a la derecha, pulsa el botón “Log out” para cerrar la sesión como administrador.



## Guía de usuario

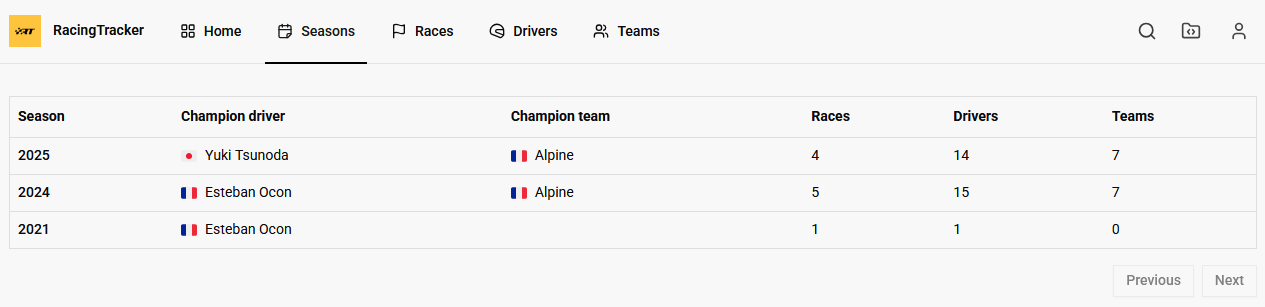
### Ver información general

Desde la página “Home”.



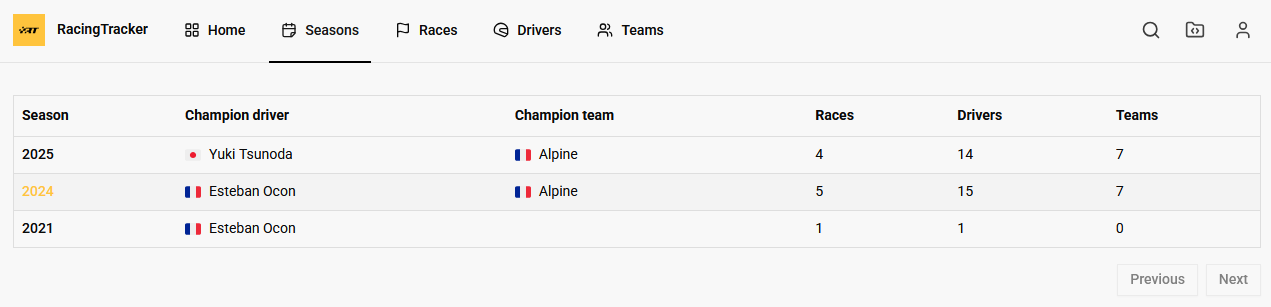
### Ver información de las temporadas

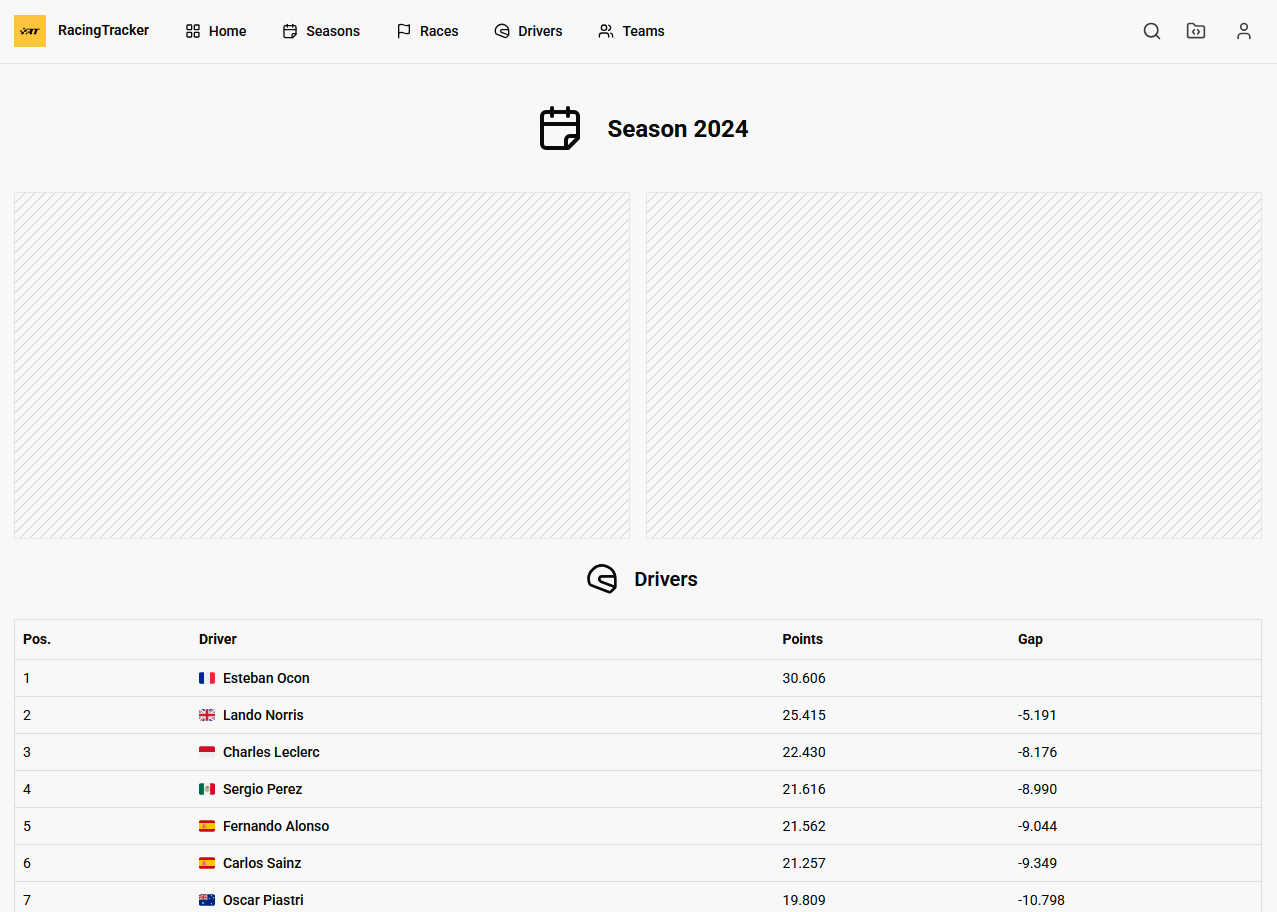
Desde la página “Seasons”.



### Ver información de una temporada

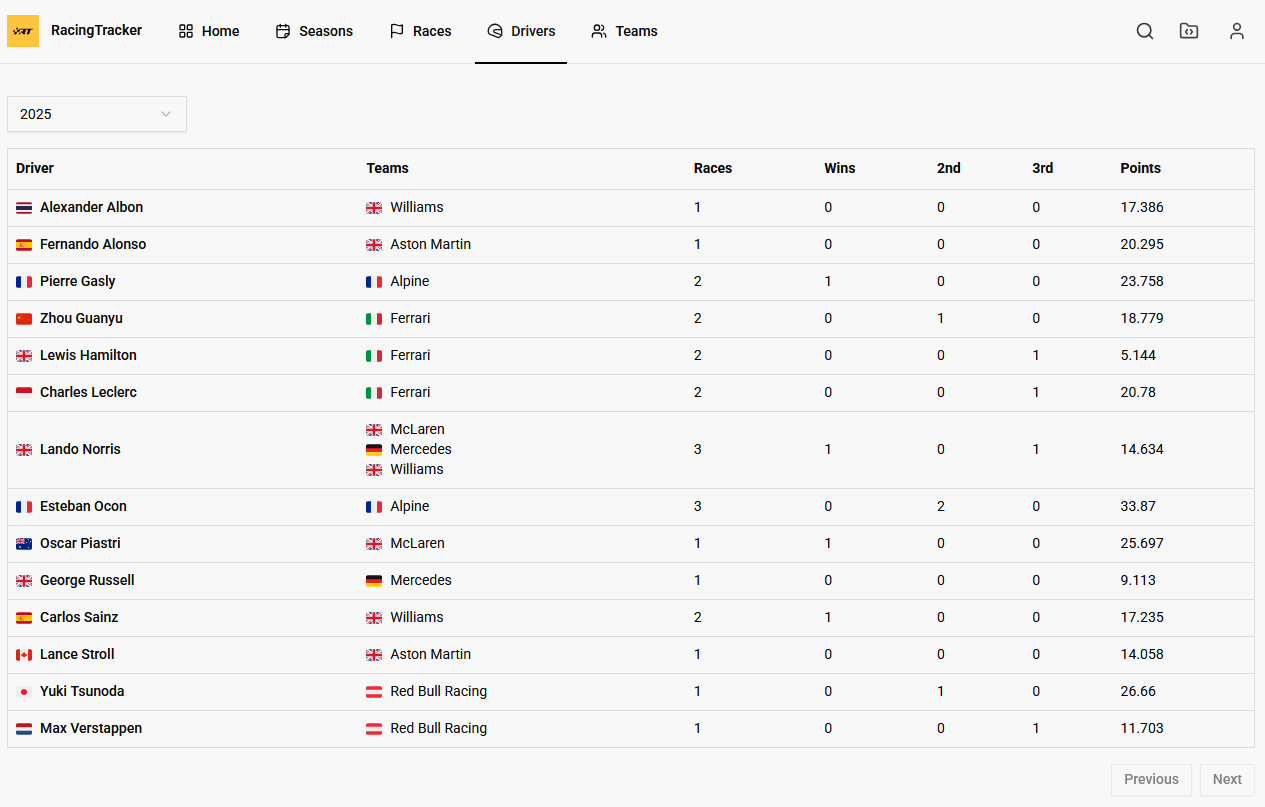
Desde la página “Seasons”, pulsa en el nombre de la temporada para entrar a su perfil.





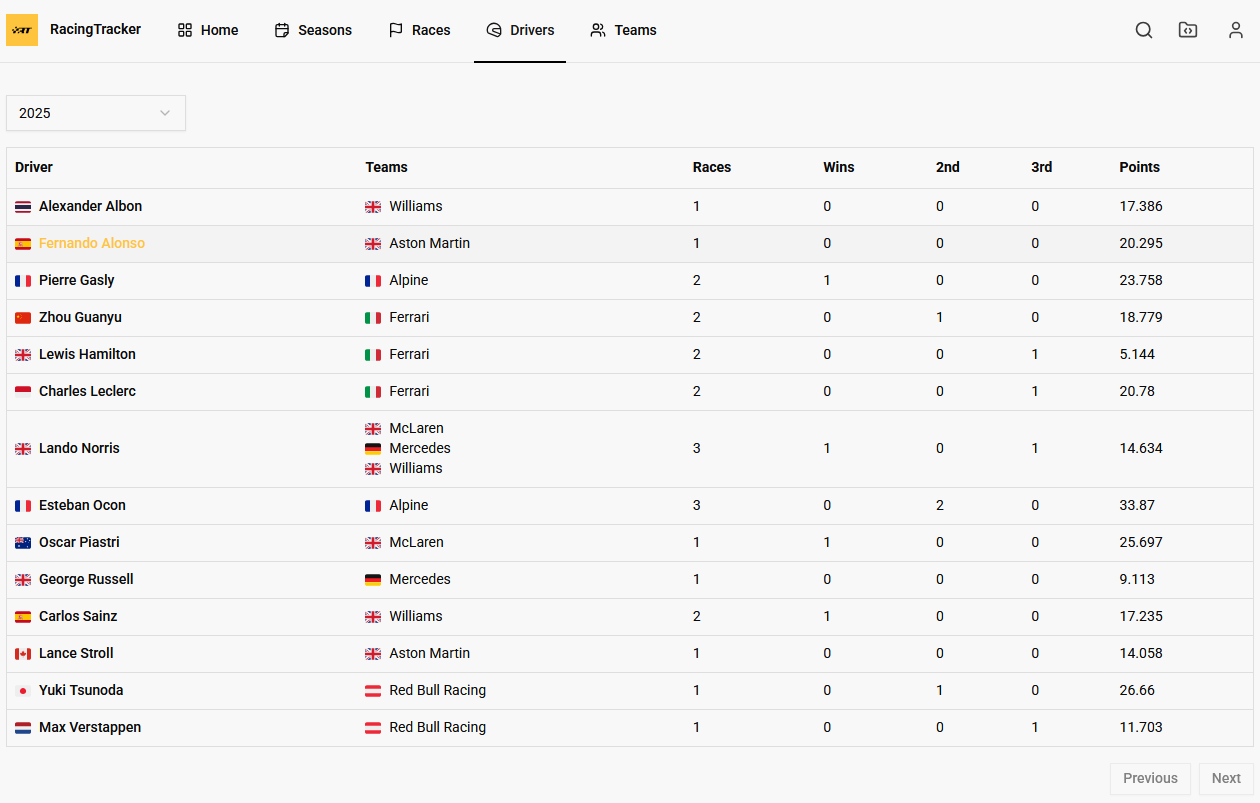
### Ver información de los pilotos

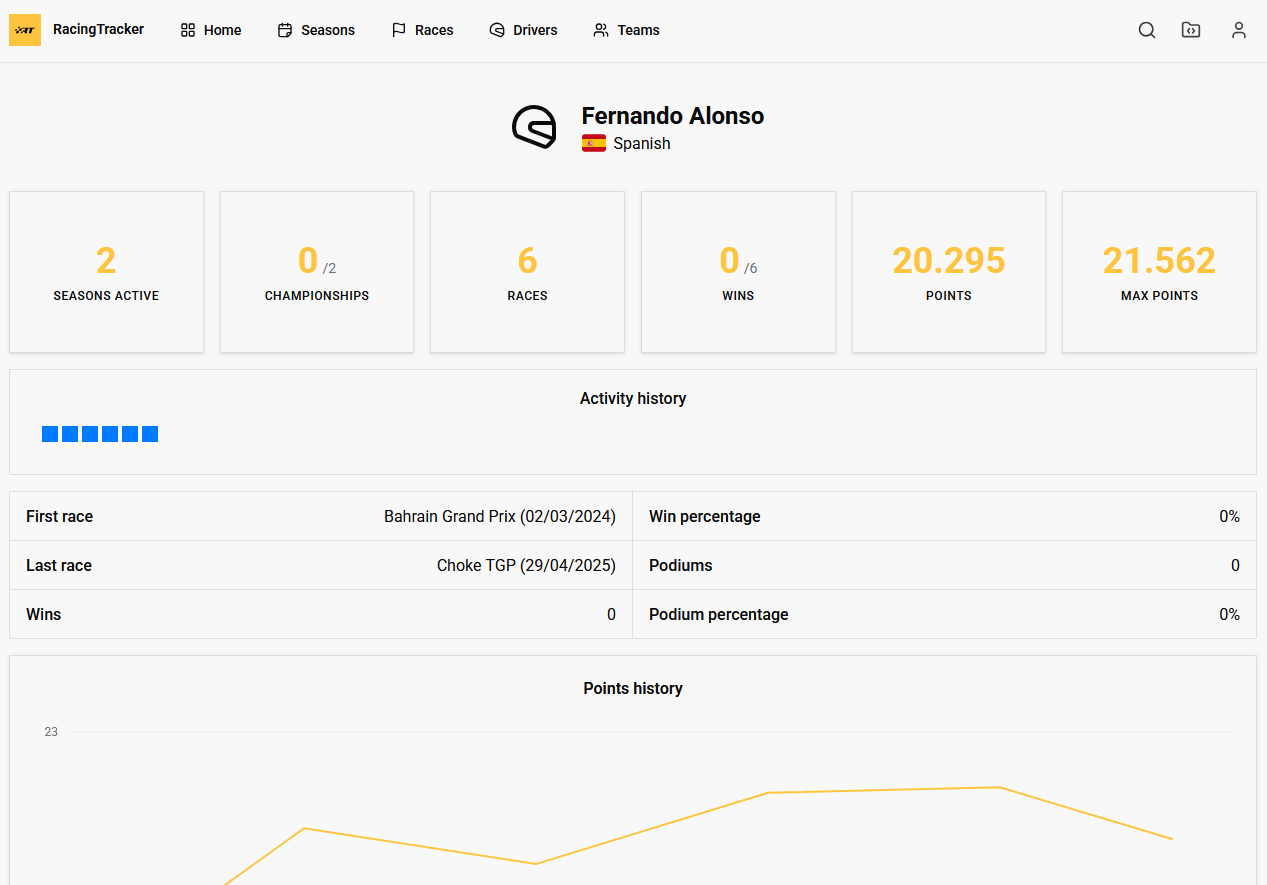
Desde la página “Drivers”.



### Ver información de un piloto

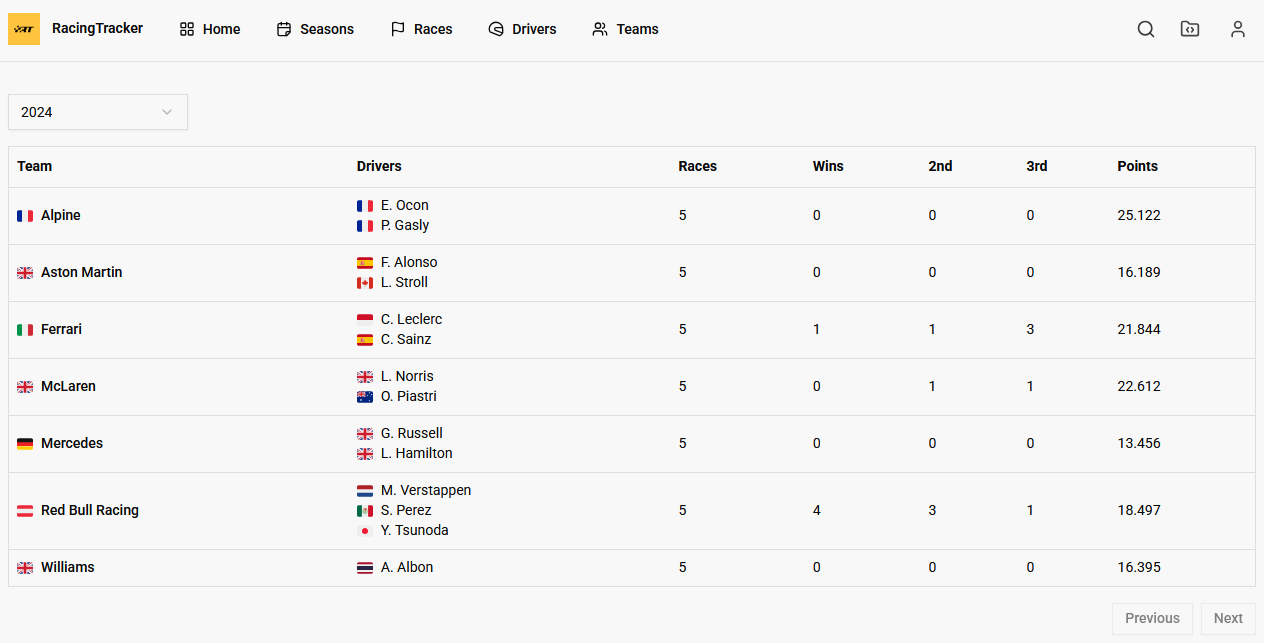
Desde la página “Drivers”, pulsa en el nombre del piloto para entrar a su perfil.





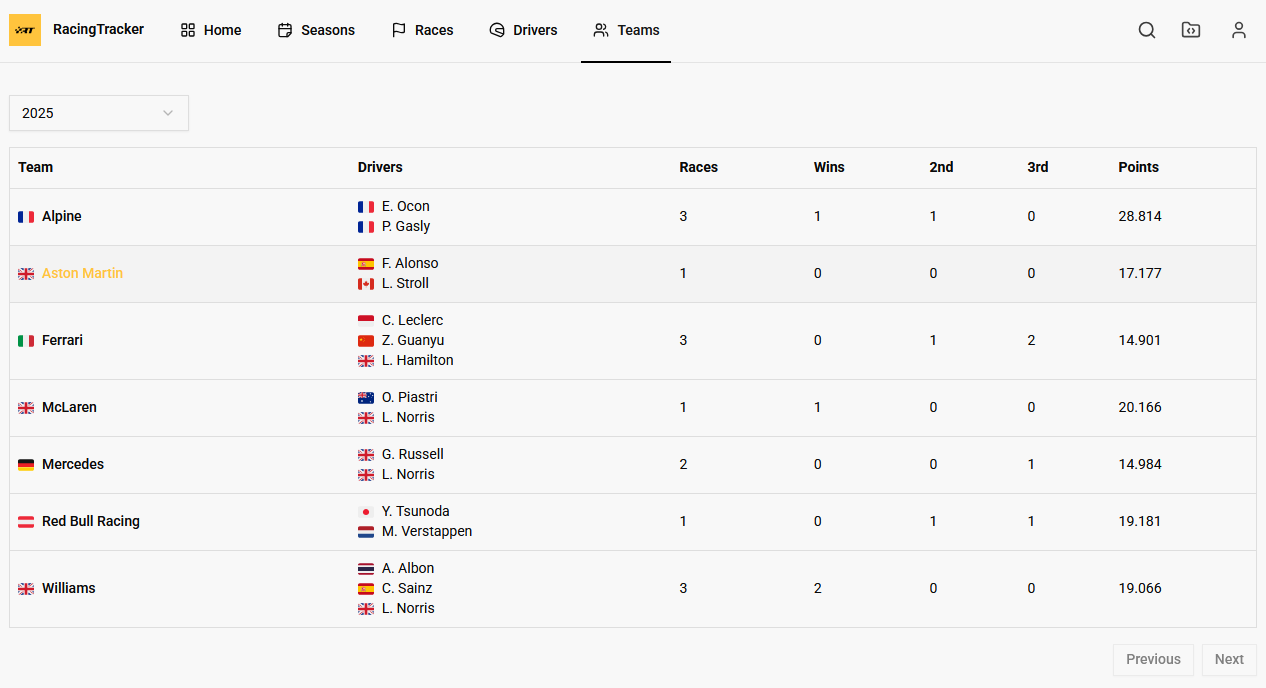
### Ver información de las escuderías

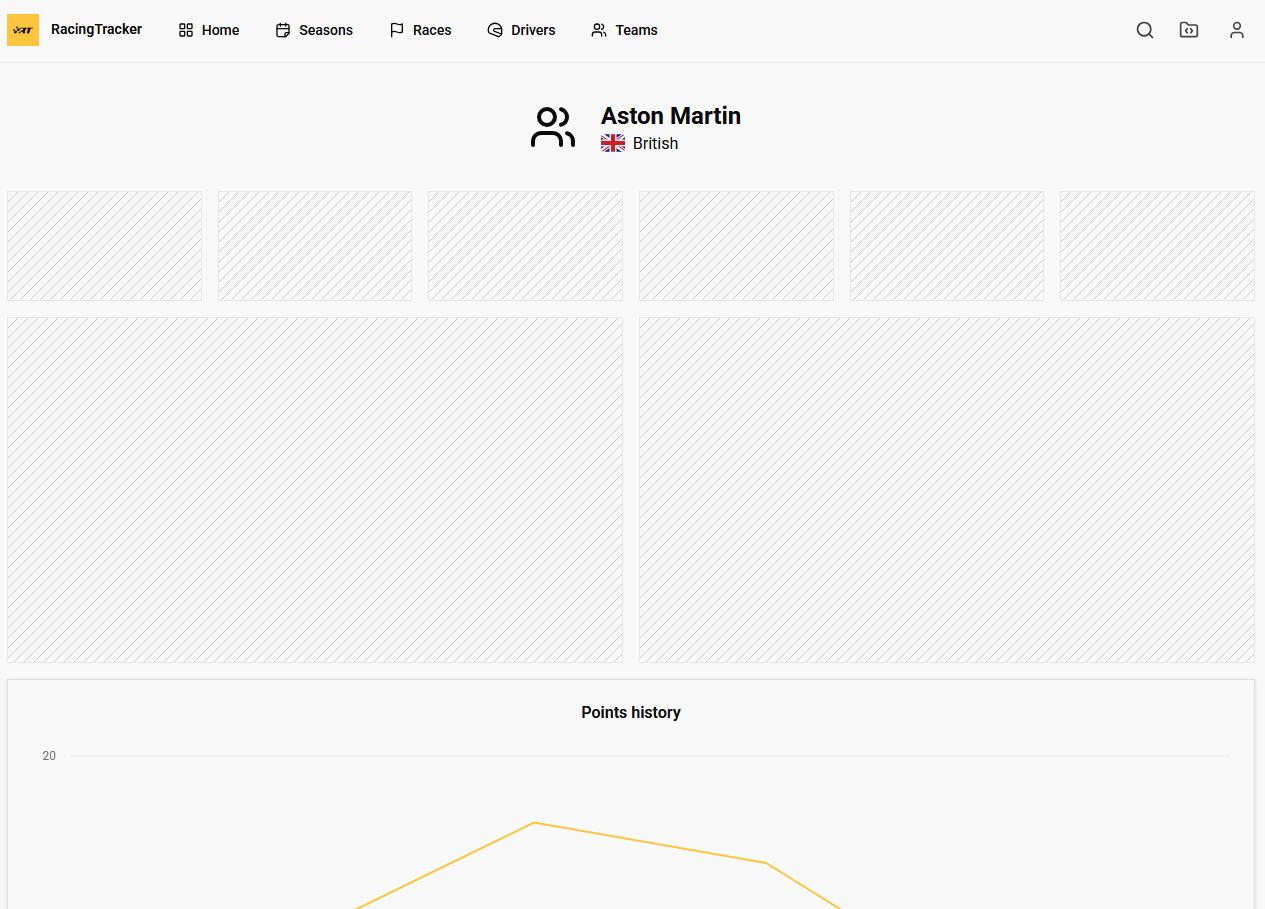
Desde la página “Teams”.



### Ver información de una escudería

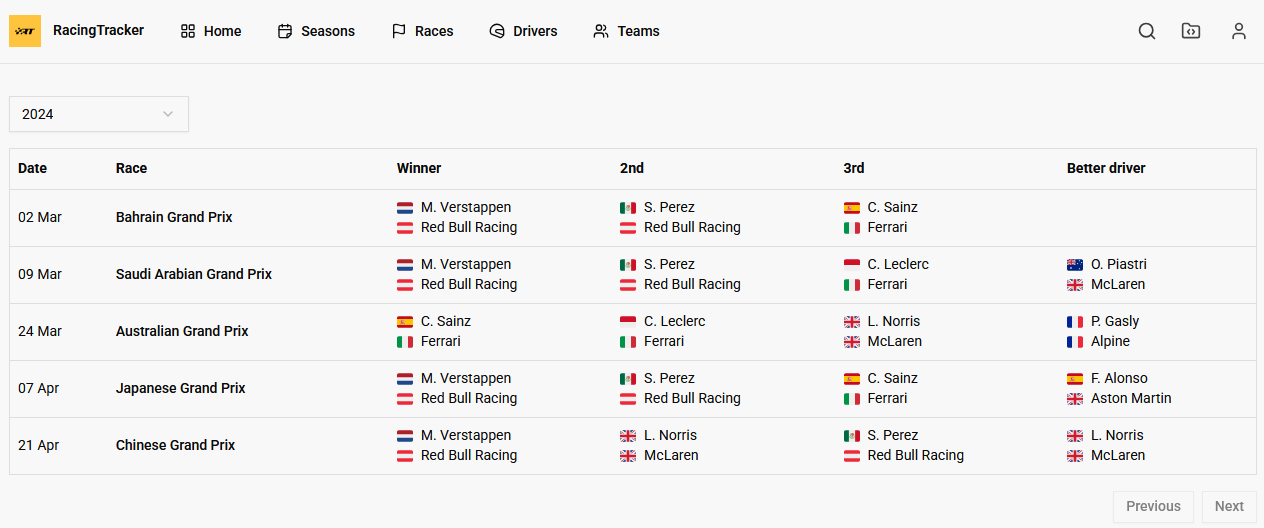
Desde la página “Teams”, pulsa en el nombre de la escudería para entrar a su perfil.





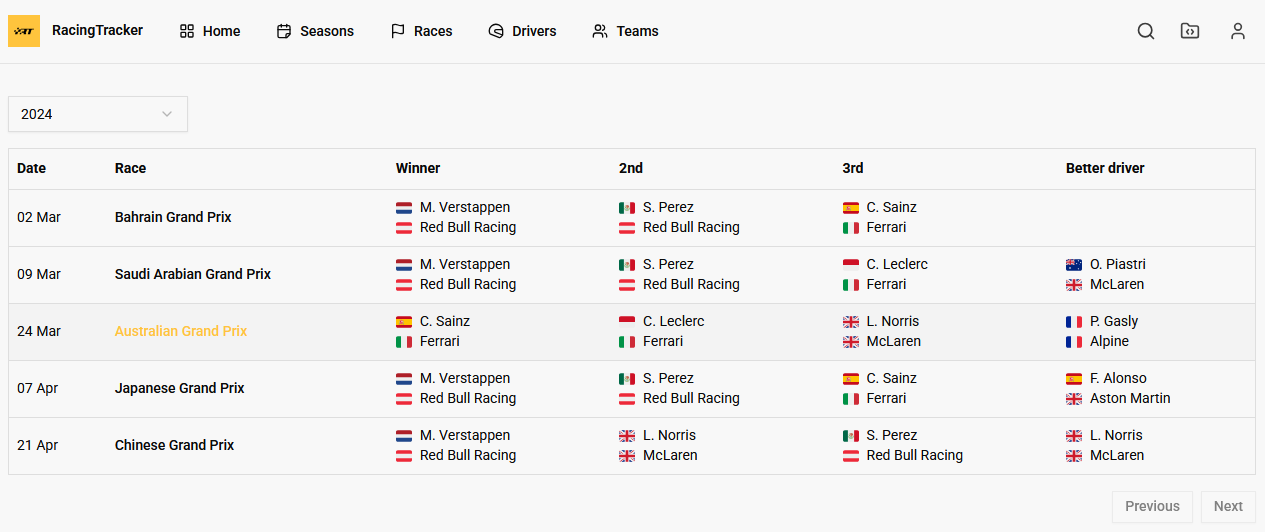
### Ver información de las carreras

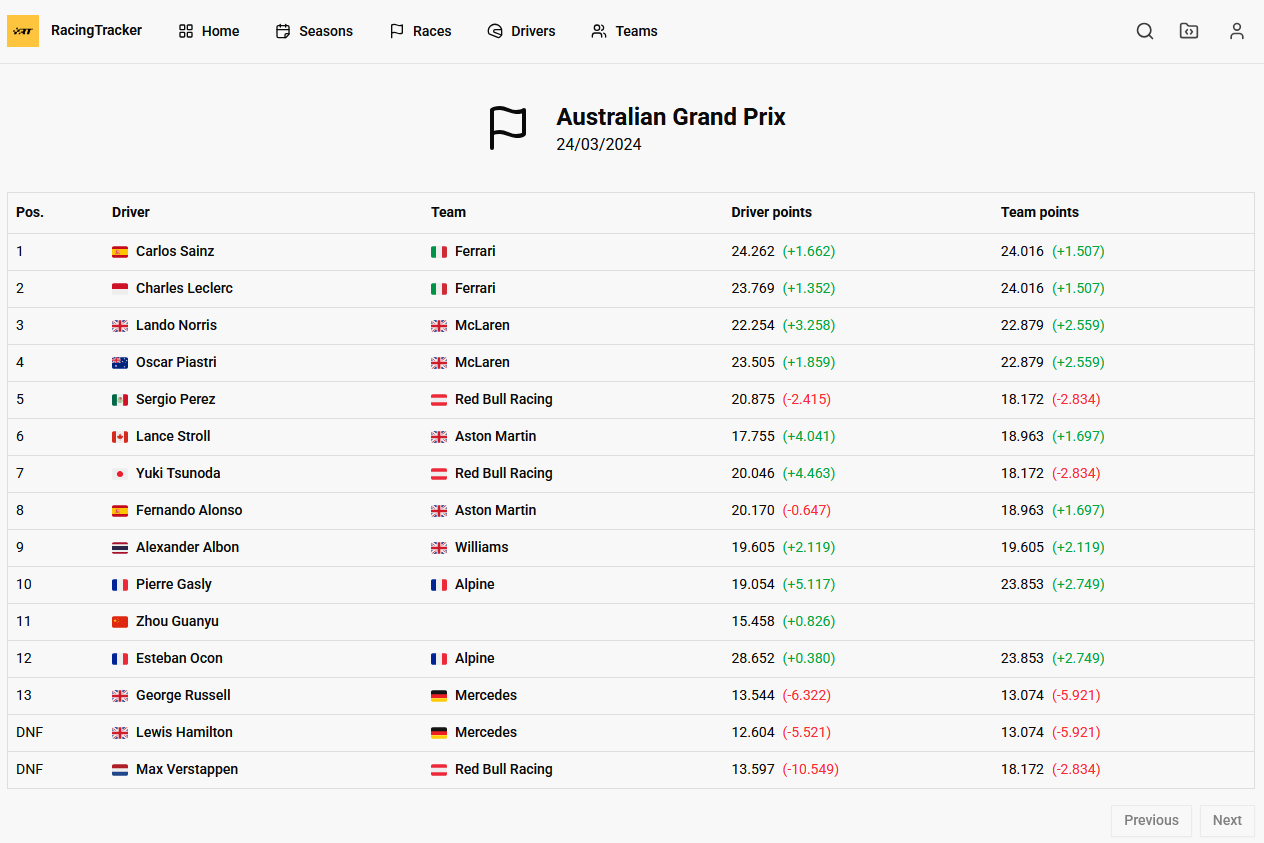
Desde la página “Races”.



### Ver información de una carrera

Desde la página “Races”, pulsa en el nombre de la carrera para entrar a su perfil.



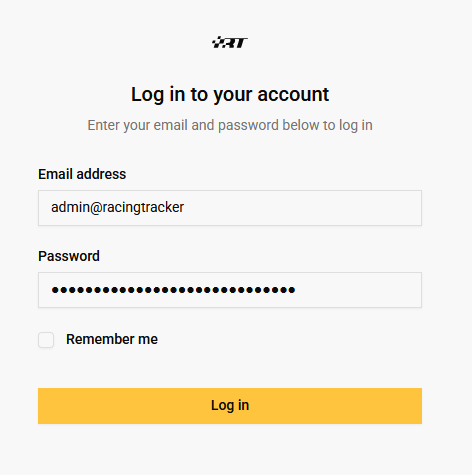


### Iniciar sesión

1. En la cabecera de la aplicación pulsa el icono de la persona, para abrir el login.



1. Escribe los datos del usuario y pulsa “Log in” para iniciar sesión.



### Abrir repositorio

Desde la cabecera de la aplicación pulsa en el icono de la carpeta para abrir en una nueva pestaña del navegador el repositorio de GitHub del proyecto.



# Bibliografía

* Editor de diagramas de base de datos → app.chartdb.io
* Editor de diagramas → drawio.com
* Diagrama de repositorio → gitdiagram.com
* Inteligencia artificial → chatgpt.com y perplexity.ai
* Editor de prototipos y diagrama de aplicación → excalidraw.com
* Iconos de banderas → svgrepo.com
* Iconos SVG → lucide.dev
* Logotipo y diagrama de Gantt → canva.com
* Componentes UI → ui.shadcn.com
* Colores y paleta → coolors.co
* Tipografía → fonts.google.com
* Proyecto que utiliza el sistema de puntuación TrueSkill → trueskill.org
* Documentación del sistema de puntuación → en.wikipedia.org/wiki/TrueSkill
* Ensayo del sistema bayesiano de calificación de habilidades TrueSkill → proceedings.neurips.cc/paper\_files/paper/2006/file/f44ee263952e65b3610b8ba51229d1f9-Paper.pdf
* Tutorial de despliegue de Laravel en AWS → [www.youtube.com/watch?v=IgJWK4aQ2jQ&list=PL1tt6av2E5dZyI15I658u\_lVhKg3jbtjS](http://www.youtube.com/watch?v=IgJWK4aQ2jQ&list=PL1tt6av2E5dZyI15I658u_lVhKg3jbtjS)
* Plantilla para la presentación → slidesgo.com