

Documentación Obligatorio

Programación I

Erika Puhl
Juan Dupont

2do Semestre 2023

1. Funcionalidades del programa

El programa es una aplicación completa de simulación de carreras de Fórmula 1 que brinda a los usuarios la capacidad de gestionar todos los aspectos relacionados con un equipo de automovilismo. Permite la creación y gestión de equipos, empleados y autos, así como la simulación de carreras con eventos imprevistos y el cálculo de resultados basado en diversos factores.

Gestión

La aplicación posibilita la creación y gestión de equipos, cada uno compuesto por pilotos, mecánicos y directores de equipo. Se ingresan detalles específicos de los empleados, como nombres, fechas de nacimiento, nacionalidades y salarios. Además, se modelan atributos exclusivos para pilotos, como su habilidad (score), número de auto, puntaje en el campeonato y estado de lesiones; y para mecánicos, como su habilidad en la preparación de autos. También se asocian modelos de autos a cada equipo para cada temporada.

Simulación de Carrera

Antes de la carrera, los usuarios pueden registrar eventos imprevistos como lesiones, abandonos, errores en pits y penalidades. La simulación utiliza una fórmula que tiene en cuenta múltiples variables, incluyendo el score del piloto, el score del auto y penalizaciones, para determinar el resultado de la carrera. Después de la carrera, se adjudican puntos a los pilotos según su posición, siguiendo las reglas del campeonato de Fórmula 1.

Consultas de Interés

La aplicación ofrece consultas estadísticas que permiten a los usuarios obtener información detallada sobre el rendimiento de los pilotos y equipos en el campeonato. Esto incluye el top 10 de pilotos con más puntos, un resumen del

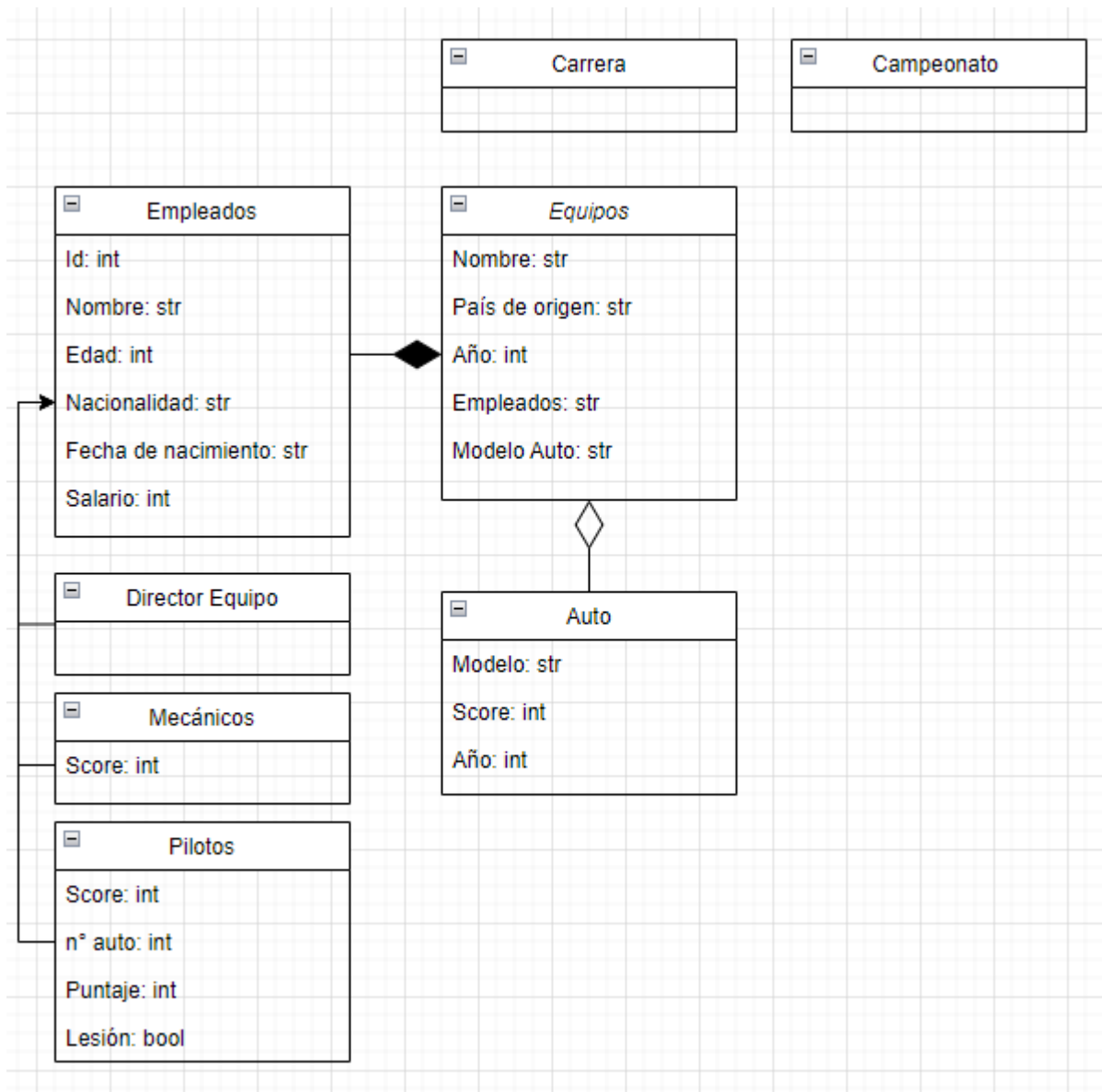
campeonato de constructores, el top 5 de pilotos mejor pagados, el top 3 de pilotos más habilidosos y la lista de jefes de equipo.

Experiencia de Usuario

La interacción con la aplicación se realiza a través de un menú en la consola de comandos, donde los usuarios pueden realizar acciones como dar de alta empleados, autos y equipos, simular carreras, realizar consultas y finalizar el programa. Se implementa una validación robusta para garantizar la entrada correcta de datos y se utilizan excepciones para manejar errores de manera amigable.

En resumen, el programa proporciona una experiencia completa de gestión y simulación de carreras de Fórmula 1, permitiendo a los usuarios sumergirse en el emocionante mundo de la competición automovilística, tomar decisiones estratégicas y evaluar el desempeño de sus equipos.

2. Diagrama UML



El diagrama UML para el simulador de carreras de Fórmula 1 se estructura alrededor de las clases fundamentales que representan los elementos clave del sistema.

En el núcleo del diseño se encuentra la clase "Carrera", que actúa como el punto focal para la simulación de carreras. Esta clase incluye métodos específicos para llevar a cabo la simulación y calcular los resultados de cada carrera individual.

A nivel más amplio, la clase "Campeonato" abarca toda la temporada de carreras, estableciendo una relación de asociación con la clase "Carrera". Esta relación indica

que un campeonato se compone de múltiples carreras, reflejando la estructura jerárquica del sistema.

La gestión de equipos se encapsula en la clase "Equipos", que representa a los participantes en el campeonato. Dentro de esta clase, se utiliza la composición para indicar que los empleados, representados por la clase abstracta "Empleado", son componentes esenciales de un equipo. Las subclases de "Empleado", como "Piloto", "Mecánicos" y "Director de Equipo", extienden la clase abstracta, estableciendo una relación de herencia de equipo.

La clase "Auto" representa los modelos de autos utilizados en cada temporada. La relación de agregación con la clase "Equipos" refleja la idea de que un equipo puede tener uno o más autos, modelando así la dinámica de la composición de autos dentro de un equipo.

En resumen, el diagrama UML proporciona una visión detallada de la estructura del simulador de carreras de F1, destacando las relaciones clave entre las clases y facilitando la comprensión y la implementación del sistema.

3. Ejecución

CODIGO

