University of British Columbia, Vancouver

Department of Computer Science

CPSC 304 Project Cover Page

Milestone #: 1

Date: 2024-10-1

Group Number: 16

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Nazia Edroos	20010476	d3e6q	edroos.nazia@gmail.com
Prajna Nayak	78725462	j8s1q	prajnapn36@gmail.com
Rachel Wang	71451769	i6a7p	rachelwang0432@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your email address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Domain Application

Domain Overview

The domain of this application is Player Statistics for Mario Kart competitions. The system will collect, organize, and analyze player performance data, providing valuable insights for competitive gameplay. This tool is essential for both players and organizers to refine their strategies and enhance overall performance metrics.

Modeled Aspects of the Domain

The database is designed to capture and analyze key aspects of player statistics within the Mario Kart competition domain. It addresses the need for players and organizers to evaluate performance based on various factors such as characters, vehicles, tracks, and power-ups used during races. For example, after a tournament, players can reference their data to see how specific combinations impacted their rankings, helping them identify which setups yield the best results. This allows for tailored strategies that can enhance competitive play.

The system is designed to serve the needs of both casual players and professional eSports competitors by providing insights into:

- **Optimal Character & Kart Combinations:** Players can analyze data to identify which combinations of characters, karts, and parts offer the best performance based on factors such as track type or player style.

- **Track Performance and Player Statistics:** The system will allow users to track individual player performance across races, measuring rankings and progress over time. These insights help players understand how they perform on specific tracks and with different combinations.

Database Specification

Database Functionality

The primary use case for this system is **eSports leagues** and competitive communities, enabling players and organizers to:

- **Optimize Gameplay**: Competitors can discover the most effective combinations of characters, karts, and parts that align with their playstyle or strategy for specific tracks.
- **Monitor and Analyze Player Performance**: Players can track their stats, including best rankings, lap times, and character usage, allowing them to make data-driven decisions in tournaments.
- **Compare Across Players**: Competitors can compare their performance with others in the league, helping to identify top players and optimal strategies for different racing conditions.

After competing in a single competition, which contains many races, players can log detailed data, including the characters they selected, the vehicles they drove, and the power-ups they deployed. This comprehensive data collection allows users to analyze how their choices impact their performance, helping them identify winning combinations and strategies.

Furthermore, the system can highlight trends in player performance over time, showcasing improvements and areas that need focus. By comparing their results against tournament winners and analyzing the effectiveness of their chosen setups, players can make informed decisions to optimize their gameplay in future competitions. This depth of analysis fosters a continuous improvement mindset and enhances competitive readiness.

By providing this level of statistical depth, the application becomes an essential tool for anyone aiming to improve their performance or organize professional-level competitions in *Mario Kart*.

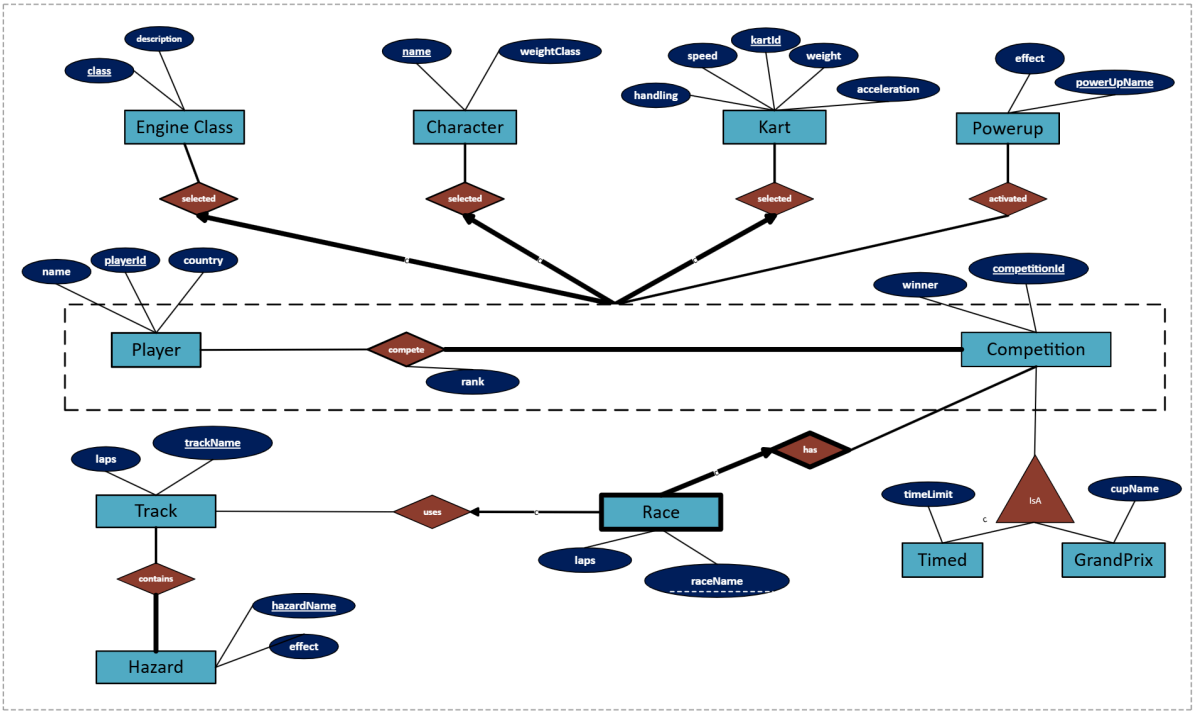
Description of the Application Platform

For this project, we will be using **Oracle** as the relational database management system (RDBMS) to manage our data, including player statistics, race details, character attributes, and track configurations. The department provides access to an Oracle server, ensuring we have the necessary support for our chosen platform.

For this project, the expected technology stack will include:

1. **JavaScript**: The primary programming language for both the frontend and backend development, allowing for a cohesive development experience across the application.
2. **Node.js**: A JavaScript runtime built on Chrome's V8 JavaScript engine, which will be used for server-side development. Node.js enables the creation of scalable network applications and facilitates handling multiple connections simultaneously.
3. **React**: A JavaScript library for building user interfaces, which will be used for the front end to create a dynamic and responsive application.

ER Diagram



ER Diagram Additional Explanation

A race is classified as a weak entity in a Mario Kart competition because it can only be uniquely identified with its associated competition. For instance, "Race 1" lacks meaning on its own, as multiple competitions can feature a "Race 1." The context of the *competitionId* is essential, distinguishing between "Competition A - Race 1" and "Competition B - Race 1." We also have an aggregation that combines the Player and Competition entities. Our goal is to treat this relationship as a single entity to accurately record the character, engine class, kart, and power-up used by a player in a specific competition. If we were to use a different design, it would reduce certain constraints and make it challenging to identify which character a player selected for a given competition. This distinction is important because players can choose multiple characters across different competitions, but only one character for each competition. By using aggregation, we are able to uniquely identify what character, powerup, kart and engine class was used in the race with the key of the aggregation.

Additionally, a race can utilize only one track type (e.g., Rainbow Road) and cannot feature multiple tracks within the same race. Conversely, a single track can be used in multiple races; for instance, both Race 1 and Race 2 can include the Rainbow Road track. We will also say that only one hazard instance can exist on a unique track to simplify our database. Furthermore, competitions can be timed, where the fastest time wins, or structured as a Grand Prix, which follows a tournament style with a chosen cup. To determine a player's ranking in each race, the relationship between the player and the race includes a rank attribute.