

NFL Player Stats

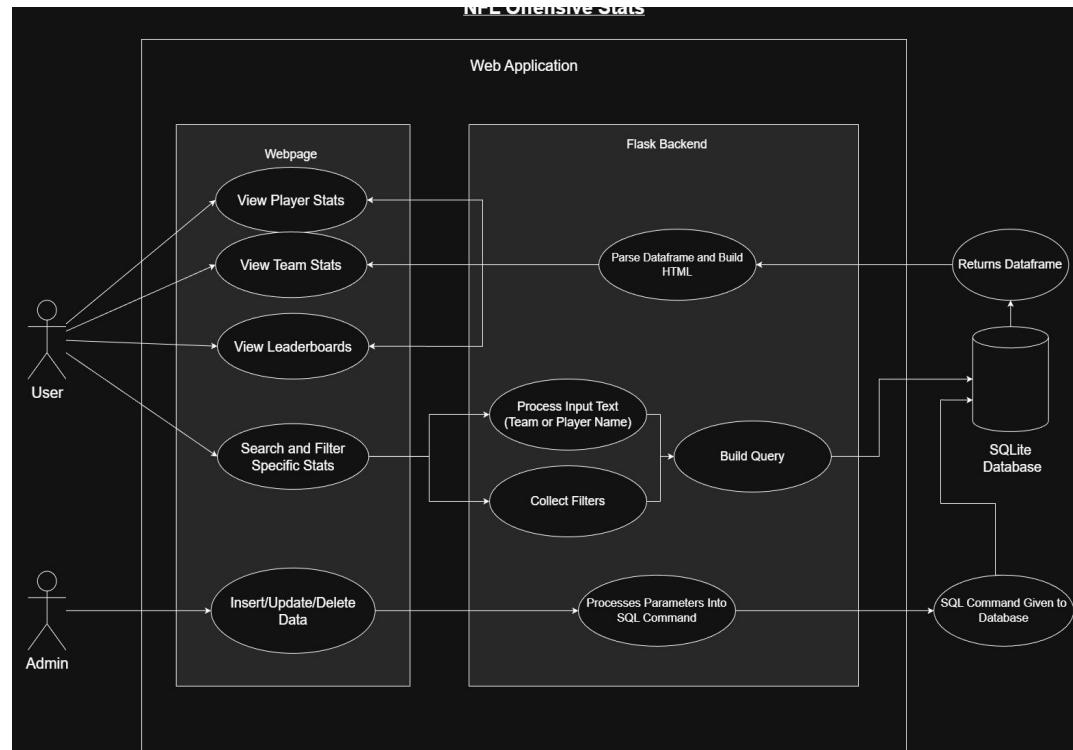
By Luther Pavline & Adrian Botello

System Description

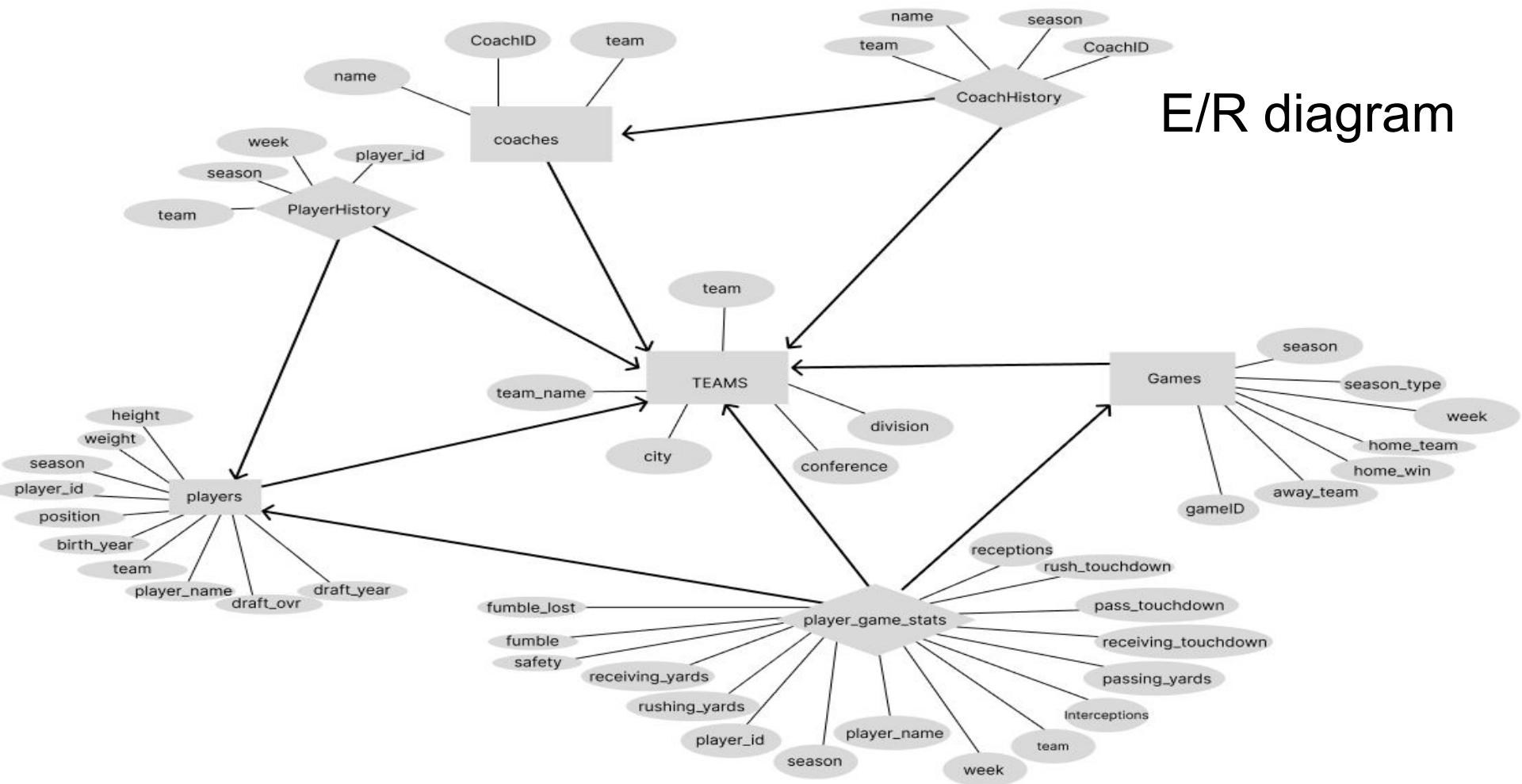
For our application we made a database that stores offensive game statistics for NFL players. It highlights key metrics, enabling users to evaluate a players performance in an effective and insightful way. In order to interact with the database, we made a web application that allows users to view individual player statistics, team statistics, and view leaderboards for various types of positions. Additionally, the administrator section allows the user to delete, edit or update data to ensure accuracy.

Use-Case description

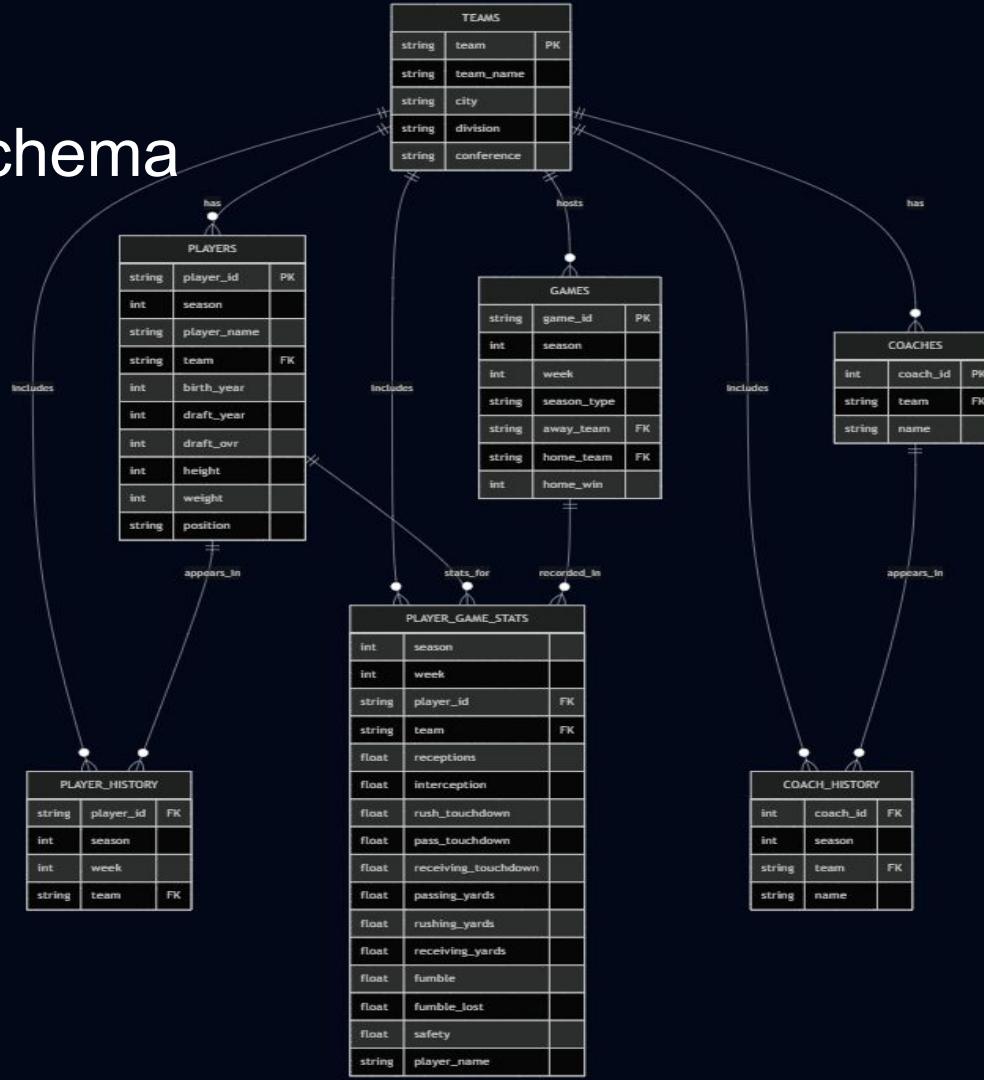
Our easiest section to navigate will be the “Leaderboards” section. The leaderboards will contain the options of choosing a season and position which will in return showcase a list of the top 10 players that have the most yards for their position. The “Player” section will take the name of a player and a list of different statistics to choose from and will return those player’s statistics along with information about the player. The “Team” section will take a team’s initial and a year and will provide a schedule and record for a team along with their total offensive yards. The website also has a toggle which switches the interface to an administrator mode which allows a record taker to modify the database by inserting, updating, and deleting data.



E/R diagram



Relational schema



Implementation Details

The application utilizes a SQLite database with Flask serving as the backend to store and manage NFL player game stats. The database is populated from a kaggle csv file containing NFL statistical data. The front end is designed using HTML, CSS, and Javascript. Flask hosts and serves these front end components, handling routing, user requests, and executing the necessary SQL queries through its Python backend to interact with the database.

Database & data: SQLite, Kaggle

Languages: Python, SQL, HTML, CSS, Javascript

Framework: Flask