

## Features

- **Add Player:** Allows the user to add a new player with a name and age.
- **View Players:** Displays a list of all players with options to edit or delete each player.
- **Edit Players:** Allows the user to edit players
- **Delete Players:** Allows the user to delete players
- **Add Player Stats:** Allows the user to add statistics (score and games played) for a selected player.
- **View Player Stats:** Displays a list of all player stats with options to edit or delete each stat.
- **Edit Stats:** Allows the user to edit statistics
- **Delete Stats:** Allows the user to delete statistics

## Prerequisites

To run this project, you will need:

- A web browser (e.g., Google Chrome, Firefox, etc.).
- A running **backend server** that exposes the necessary endpoints for managing players and stats (e.g., GET, POST, PUT, and DELETE methods for players and stats).
- The frontend is designed to interact with an API hosted at <http://localhost:8080> (make sure your backend is running at this address or update the `API_BASE_URL` in the code accordingly).

## Installation

1. Clone this repository to your local machine:

```
git clone https://github.com/mitutib/PlayerStatsApp.git
```

2. Navigate to the project directory:

```
cd PlayerStatsApp
```

3. Open the `index.html` file in your preferred web browser.

Alternatively, you can run a local server (using tools like VSCode's Live Server extension or any other HTTP server) to view the application in a development environment.

## File Structure

The project consists of the following key files:

- `index.html`: The main HTML file for the application. It includes the layout and structure.
- `style.css`: The custom CSS file for styling the frontend.
- `script.js`: The JavaScript file responsible for fetching and displaying player data, handling player actions (add, edit, delete), and managing player stats.
- <https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css>: Bootstrap 5 CSS used for responsive design and layout.
- <https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js>: Bootstrap JS for additional UI functionality.

## Usage

1. **Add Player:**
  - a. Fill out the player name and age in the provided form and click the "Add Player" button.
  - b. The new player will appear in the list of players, and you can edit or delete them.
2. **Add Player Stats:**
  - a. Select a player from the dropdown list.
  - b. Enter the player's score and the number of games played, then click the "Add Stats" button.
  - c. The stats will be added to the player's record and displayed in the stats list.
3. **Edit Player/Stats:**
  - a. Click the "Edit" button next to a player or stat to modify the player's name, age, score, or games played.
4. **Delete Player/Stats:**
  - a. Click the "Delete" button next to a player or stat to remove them from the list.

## API Endpoints

The frontend communicates with the backend API through the following endpoints:

- **GET /players:** Fetches the list of players.
- **POST /players:** Adds a new player.
- **PUT /players/{id}:** Edits the details of an existing player.
- **DELETE /players/{id}:** Deletes a player.
- **GET /stats:** Fetches the list of player statistics.
- **POST /stats:** Adds statistics for a player.
- **PUT /stats/{id}:** Edits an existing stat.
- **DELETE /stats/{id}:** Deletes a stat.