ADVANCE DATABASE SYSTEMS DESIGN FINAL PROJECT
SOCCER WEB APP USING BOTTLE AND SQL
NAME: **PARITOSH GANDRE** STUDENT ID: **811292707**
GITHUB REPO LINK: **https://github.com/ParitoshKent/adsd-project**

SOCCER WEB APP

1. **What is my project?**
    - This project is about sports data management, which is based on Soccer data.
    - Soccer web application is an application that can be used in sports data management. This type of application helps in keeping track of player and team data.
2. **Why is it useful?**
    - It can be used to store player's information which is very important for any club/team.
    - **Players Page:** Displays a list of players with their names, positions, and respective teams. Provides options to update player information or delete players.
    - **Teams Page:** Lists all soccer teams with their names and team IDs.
    - **Add Player Page:** Allows the addition of new players by entering player name, and position, and selecting the team.
    - **Search Players Page:** A dedicated page for searching players by name. Displays search results with player names, positions, and team names.
3. **Why does this application require two tables?**
    - To effectively organize and handle data, the Soccer Manager web app needs at least two tables: one for players and one for teams.
    - This design decision guarantees a standardized structure, makes it easier to understand the links between teams and players, preserves data integrity, and permits manageable and scalable database administration.
    - Tables for players and teams are kept separate to improve querying, reporting, and general application flexibility.
4. **Explaining the decisions you made in choosing your database and access method**
    - The desire for simplicity and setup convenience for effective data management led to the selection of SQL, more especially SQLite.
    - Because of their easy-to-use interface and minimal configuration requirements, SQL databases are suitable for developers with different levels of experience.

SOURCE CODE

- **Setup.py**

```
import sqlite3

db_path = 'soccer.db'
def create_tables():
    conn = sqlite3.connect(db_path)
    cursor = conn.cursor()
    # Create the 'teams' table
    cursor.execute('''
        CREATE TABLE IF NOT EXISTS teams (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            name TEXT NOT NULL
        )
    ''')
    # Create the 'players' table with a foreign key relationship
    cursor.execute('''
        CREATE TABLE IF NOT EXISTS players (
            id INTEGER PRIMARY KEY AUTOINCREMENT,
            name TEXT NOT NULL,
            position TEXT NOT NULL,
            team_id INTEGER,
            FOREIGN KEY (team_id) REFERENCES teams(id)
        )
    ''')
    conn.commit()
    conn.close()
def insert_initial_data():
    conn = sqlite3.connect(db_path)
    cursor = conn.cursor()
    # Insert sample teams
    cursor.execute("INSERT INTO teams (name) VALUES ('Team A')")
    cursor.execute("INSERT INTO teams (name) VALUES ('Team B')")
    cursor.execute("INSERT INTO teams (name) VALUES ('Team C')")
    # Insert sample players with team_id
    cursor.execute("INSERT INTO players (name, position, team_id) VALUES ('Player 1',
'Forward', 1)")
    cursor.execute("INSERT INTO players (name, position, team_id) VALUES ('Player 2',
'Midfielder', 2)")
    cursor.execute("INSERT INTO players (name, position, team_id) VALUES ('Player 3',
'Defender', 3)")
    conn.commit()
    conn.close()
```

```
if __name__ == '__main__':
    create_tables()
    insert_initial_data()
    print('Database setup complete.')
```

- **Application.py:**

```
from bottle import Bottle, template, request, redirect,route,run
import sqlite3

conn = sqlite3.connect("soccer.db")
cursor = conn.cursor()

@route('/')
def home():
    return template('home')

@route('/players')
def players():
    cursor.execute("SELECT players.id, players.name, players.position, teams.name FROM
players JOIN teams ON players.team_id = teams.id")
    players = cursor.fetchall()
    return template('players', players=players)

@route('/teams')
def teams():
    cursor.execute("SELECT * FROM teams")
    teams = cursor.fetchall()
    return template('teams', teams=teams)

@route('/add_player', method='GET')
def add_player_form():
    cursor.execute("SELECT * FROM teams")
    teams = cursor.fetchall()
    return template('add_player', teams=teams)

@route('/add_player', method='POST')
def add_player():
    player_name = request.forms.get('player_name')
    position = request.forms.get('position')
    team_id = request.forms.get('team_id')
```

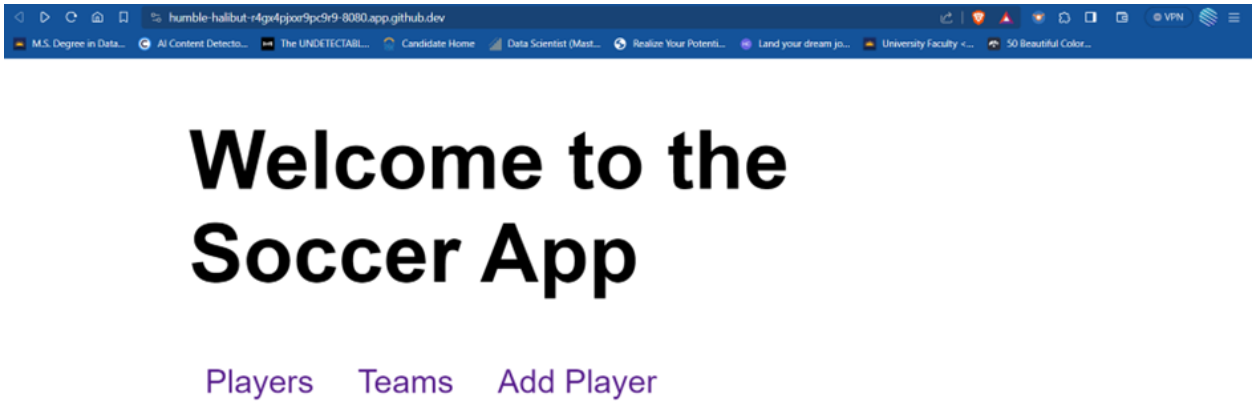
```
    cursor.execute("INSERT INTO players (name, position, team_id) VALUES (?, ?, ?)",
(player_name, position, team_id))
    conn.commit()

    redirect('/players')

@route('/update_player/<player_id>', method='GET')
def update_player_form(player_id):
    cursor.execute("SELECT * FROM players WHERE id=?", (player_id,))
    player = cursor.fetchone()

    cursor.execute("SELECT * FROM teams")
    teams = cursor.fetchall()

    return template('update_player', player=player, teams=teams)

@route('/update_player/<player_id>', method='POST')
def update_player(player_id):
    player_name = request.forms.get('player_name')
    position = request.forms.get('position')
    team_id = request.forms.get('team_id')

    cursor.execute("UPDATE players SET name=?, position=?, team_id=? WHERE id=?",
(player_name, position, team_id, player_id))
    conn.commit()

    redirect('/players')

@route('/delete_player/<player_id>')
def delete_player(player_id):
    cursor.execute("DELETE FROM players WHERE id=?", (player_id,))
    conn.commit()

    redirect('/players')

@route('/search_players')
def search_players_form():
    return template('search_players', players=None, search_query=None)

@route('/search_players', method='get')
def search_players():
    search_query = request.query.get('search', '').strip()

    if search_query:
```

```
        cursor.execute("SELECT players.name, players.position, teams.name FROM players JOIN
teams ON players.team_id = teams.id WHERE players.name LIKE ?", ('%' + search_query +
'%',))
        players = cursor.fetchall()
    else:
        players = None

    return template('search_players', players=players, search_query=search_query)

if __name__ == '__main__':
    run(host='localhost', port=8080, debug=True)
```

- **views/home.tpl:**

```
<!DOCTYPE html>
<html>
<head>
<style>
body{
        width:70%;
        font-size:50px;
        margin:0 auto;
    }
    a{
        font-size:40px;
        margin:20px;
        text-decoration:none;
    }
    table{
        padding:10px;
    }
    tr td{
        padding:10px;
    }
</style>
    <title>Soccer App</title>
</head>
<body>
    <h1>Welcome to the Soccer App</h1>
    <a href="/players">Players</a>
    <a href="/teams">Teams</a>
    <a href="/add_player">Add Player</a>
    <a href="/search_players">Search Players</a>
</body>
</html>
```

Displaying Home page of the Soccer web application.

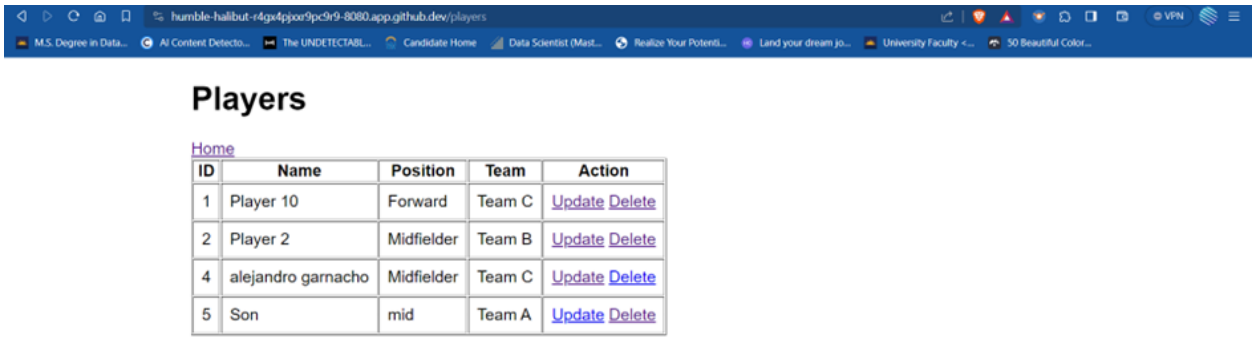


- **views/players.tpl**

```
<!DOCTYPE html>
<html>
<head>
<style>
body{
        width:70%;
        font-size:20px;
        margin:0 auto;
    }
    td{
        padding:10px;
    }

</style>
    <title>Players</title>
</head>
<body>
    <h1>Players</h1>
    <a href="/" >Home</a>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Position</th>
```

```
            <th>Team</th>
            <th>Action</th>
        </tr>
        % for player in players:
            <tr>
                <td>{{ player[0] }}</td>
                <td>{{ player[1] }}</td>
                <td>{{ player[2] }}</td>
                <td>{{ player[3] }}</td>
                <td>
                    <a href="/update_player/{{ player[0] }}">Update</a>
                    <a href="/delete_player/{{ player[0] }}">Delete</a>
                </td>
            </tr>
        % end
    </table>
<a href="/add_player">Add Player</a>

</body>
</html>
```

Displaying Players list from the database with Team column from “Teams” table:

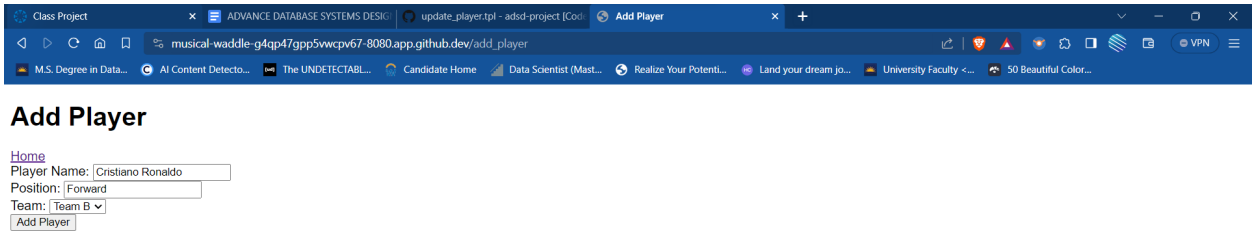


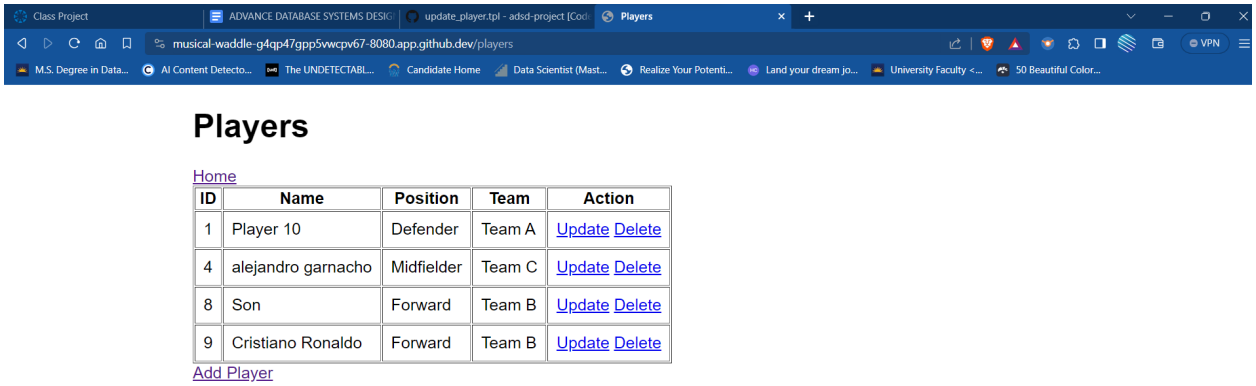
- **views/add_player.tpl**

```
<!DOCTYPE html>
<html>
<head>
    <title>Add Player</title>
</head>
<body>
    <h1>Add Player</h1>
```

```
    <a href="/">Home</a>
    <form action="/add_player" method="post">
        <label for="player_name">Player Name:</label>
        <input type="text" name="player_name" required><br>

        <label for="position">Position:</label>
        <input type="text" name="position" required><br>

        <label for="team_id">Team:</label>
        <select name="team_id" required>
            % for team in teams:
                <option value="{{ team[0] }}">{{ team[1] }}</option>
            % end
        </select><br>
        <input type="submit" value="Add Player">
    </form>
</body>
</html>
```

Adding “Cristiano Ronaldo” to the database:





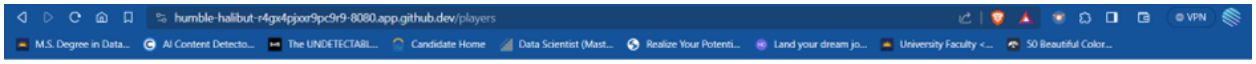
Players

Home

ID	Name	Position	Team	Action
1	Player 10	Defender	Team A	Update Delete
4	alejandro garnacho	Midfielder	Team C	Update Delete
8	Son	Forward	Team B	Update Delete
9	Cristiano Ronaldo	Forward	Team B	Update Delete

Add Player

Deleting “Player 2” from the database:



Players

Home

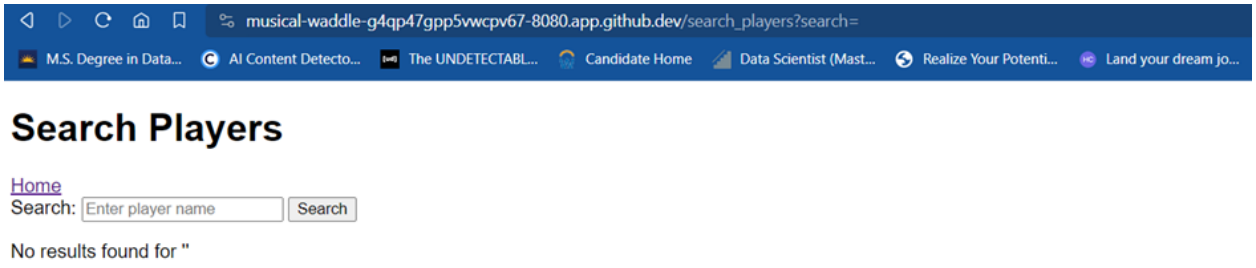
ID	Name	Position	Team	Action
1	Player 10	Defender	Team A	Update Delete
4	alejandro garnacho	Midfielder	Team C	Update Delete
5	Son	mid	Team A	Update Delete

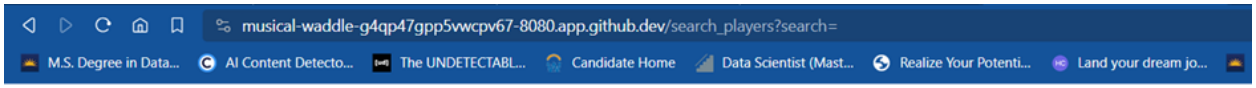
- **views/search_players.tpl:**

```
<!DOCTYPE html>
<html>
<head>
    <title>Search Players</title>
</head>
<body>
    <h1>Search Players</h1>
    <a href="/">Home</a>
    <form action="/search_players" method="get">
        <label for="search">Search:</label>
        <input type="text" name="search" placeholder="Enter player name">
        <input type="submit" value="Search">
    </form>
```

```
    % if players:
        <table border="1">
            <tr>
                <th>Name</th>
                <th>Position</th>
                <th>Team</th>
                <th>Action</th>
            </tr>
            % for player in players:
                <tr>
                    <td>{{ player[0] }}</td>
                    <td>{{ player[1] }}</td>
                    <td>{{ player[2] }}</td>
                    <td>
                        <a href="/update_player/{{ player[0] }}">Update</a>
                        <a href="/delete_player/{{ player[0] }}">Delete</a>
                    </td>
                </tr>
            % end
        </table>
    % else:
        <p>No results found for '{{ search_query }}'</p>
    % end
</body>
</html>
```

Searching for “Alejandro”:

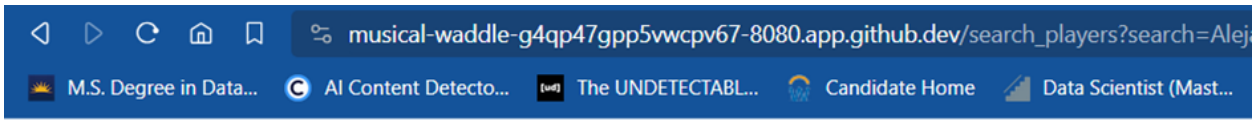




Search Players

Home
Search: Alejandro Search

No results found for ''



Search Players

Home
Search: Enter player name Search

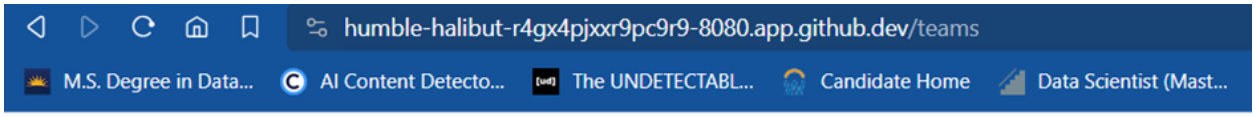
Name	Position	Team	Action
alejandro garnacho	Midfielder	Team C	Update Delete

- **views/teams.tpl:**

```
<!DOCTYPE html>
<html>
<head>
    <title>Teams</title>
</head>
<body>
    <h1>Teams</h1>
    <a href="/">Home</a>
    <table border="1">
        <tr>
            <th>Team ID</th>
            <th>Team Name</th>
        </tr>
        % for team in teams:
            <tr>
                <td>{{ team[0] }}</td>
                <td>{{ team[1] }}</td>
            </tr>
```

```
        % end
    </table>
</body>
</html>
```

Displaying Teams database:



Teams

Home

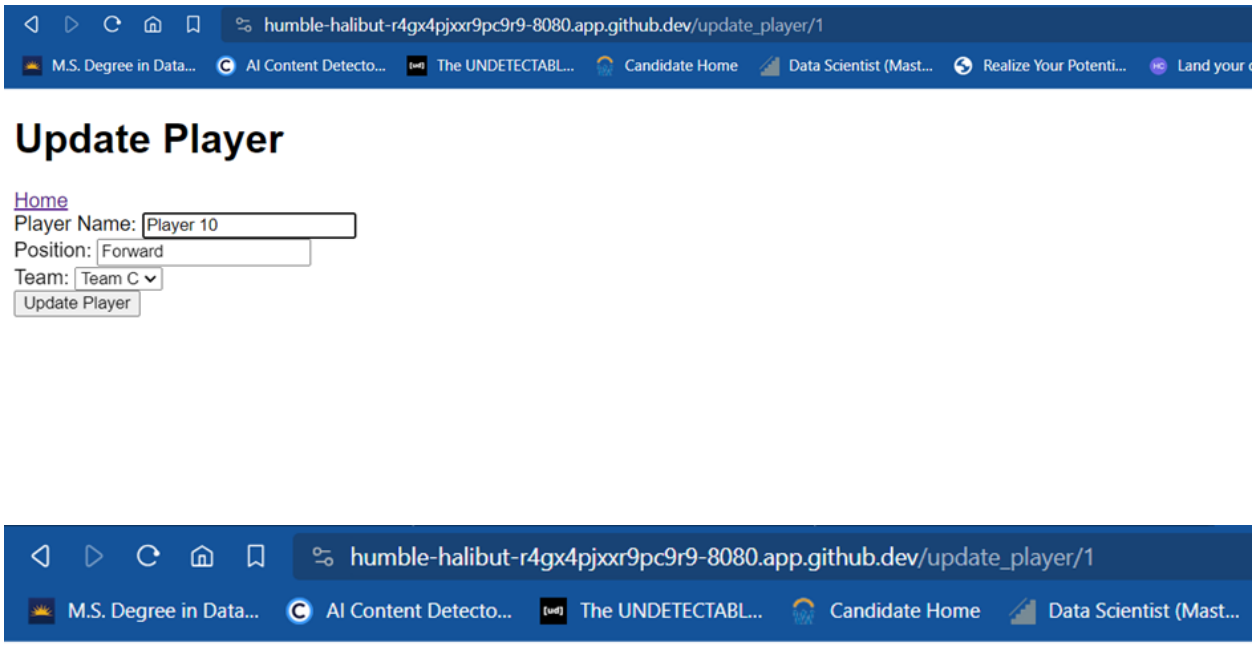
Team ID	Team Name
1	Team A
2	Team B
3	Team C

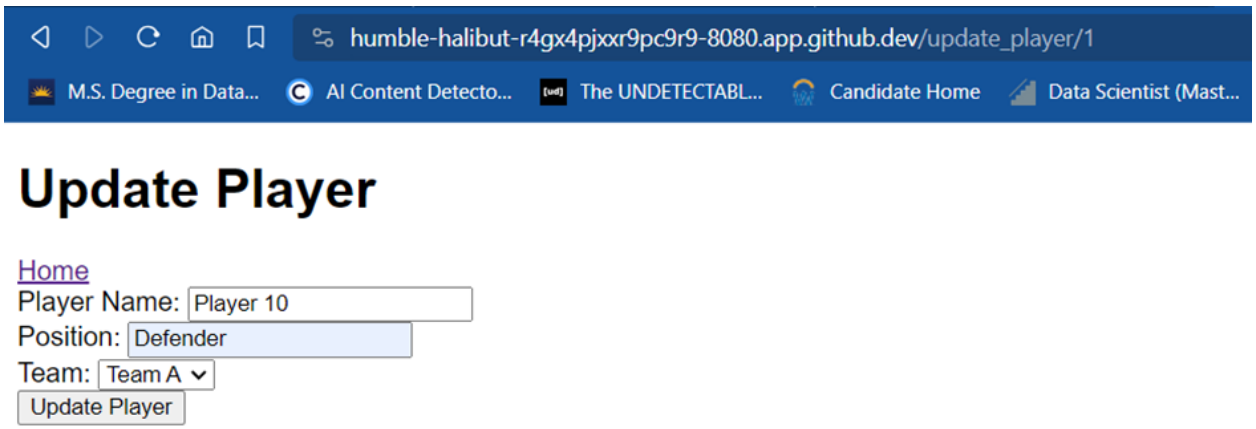
- **views/update_player.tpl:**

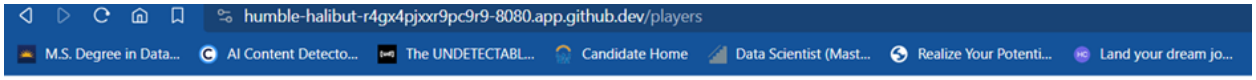
```
<!DOCTYPE html>
<html>
<head>
    <title>Update Player</title>
</head>
<body>
    <h1>Update Player</h1>
    <a href="/">Home</a>
    <form action="/update_player/{{ player[0] }}" method="post">
        <label for="player_name">Player Name:</label>
        <input type="text" name="player_name" value="{{ player[1] }}" required><br>

        <label for="position">Position:</label>
        <input type="text" name="position" value="{{ player[2] }}" required><br>

        <label for="team_id">Team:</label>
        <select name="team_id" required>
            % for team in teams:
                <option value="{{ team[0] }}" % if team[0] == player[3]: selected % end>{{ team[1]
}}</option>
            % end
        </select><br>
```

```
        <input type="submit" value="Update Player">
    </form>
</body>
</html>
```

Updating “Player 10”s Position from Forward to Defender:







Players

Home

ID	Name	Position	Team	Action
1	Player 10	Defender	Team A	Update Delete
2	Player 2	Midfielder	Team B	Update Delete
4	alejandro garnacho	Midfielder	Team C	Update Delete
5	Son	mid	Team A	Update Delete