

Course Title:	Software Design Architecture
Course Number:	COE692
Semester/Year	W2025

Instructor:	Dr. Faezeh Ensan
--------------------	-------------------------

<i>Assignment/Lab Number:</i>	Lab 2
<i>Assignment/Lab Title:</i>	Employing n-Tier Architecture for creating a Web application for the specified project

<i>Submission Date:</i>	February 21, 2024 10:00 PM
<i>Due Date:</i>	February 21, 2025 11:59 PM

Student LAST Name	Student FIRST Name	Student Number	Section	Signature*
Kuganendra	Aananth	501182297	2	AK
Sethi	Ayaan	501179118	2	AS

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at:
<https://www.torontomu.ca/content/dam/senate/policies/pol60.pdf>

Part 1: ER Diagram and SQL Tables

The first stages of the lab requires data modelling through an ER diagram, the creation of the database, and the insertion of data using SQL statements. Figure 1 displays the ER model which displays attributes and relationships for the four main entities in this web application: League, Manager, Team, and Player.

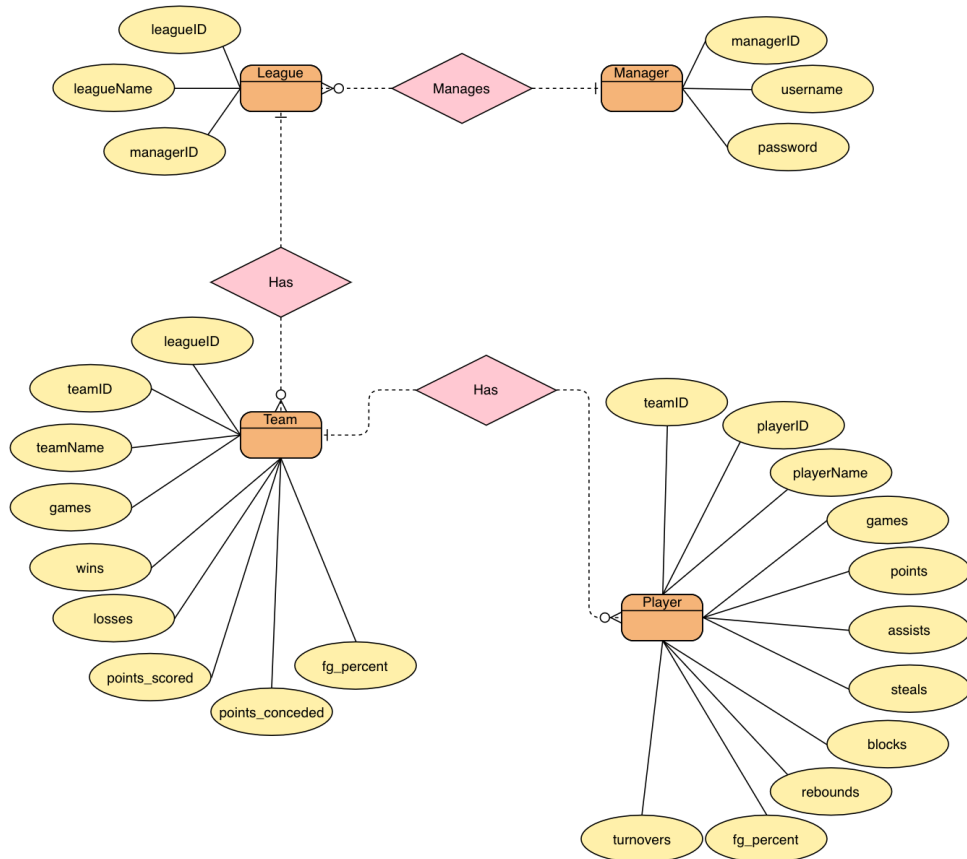


Figure 1: ER Model

Description

The ER diagram has four entities: League, Manager, Team, and Player. There is a one-to-many relationship between the Manager and League since one manager can manage many leagues. However, each league must only have one manager. The League has attributes, leagueID (key); leagueName; and managerID (foreign key from manager), to support the creation of different leagues under the same or different managers.

The manager also has three attributes, managerID (key); username; password, where the manager can be easily identified and authentication keys are stored.

The League has a one-to-many relationship with the Team entity. There are many teams in a league, but each team can only have one league. The Team entity has the following attributes: leagueID (foreign key from League), teamID (key), teamname, games, wins, losses, points_scored, points_conceded, and fg_percent.

The Team has a one-to-many relationship with the Player entity. There are many players in a team, but each player can only have one team. The Player entity has the following attributes: teamID (foreign key from Team), playerID (key), playername, games, points, assists, steals, blocks, rebounds, fg_percent, and turnovers.

The SQL code for the creation of tables from the ER diagram in Figure 1.0 is shown below.

```
CREATE TABLE Manager (  
    managerID INT AUTO_INCREMENT PRIMARY KEY,  
    username VARCHAR(30) UNIQUE NOT NULL,  
    password VARCHAR(30) NOT NULL  
);
```

Figure 1.1: SQL Commands for table of creation of Manager entity.

```
CREATE TABLE League (  
    leagueID INT AUTO_INCREMENT PRIMARY KEY,  
    leagueName VARCHAR(50) NOT NULL UNIQUE,  
    managerID INT NOT NULL,  
    FOREIGN KEY (managerID) REFERENCES Manager(managerID) ON DELETE CASCADE  
);
```

Figure 1.2: SQL Commands for table of creation of League entity.

```
CREATE TABLE Team (  
    teamID INT AUTO_INCREMENT PRIMARY KEY,  
    teamName VARCHAR(75) NOT NULL,  
    leagueID INT NOT NULL,  
    games INT DEFAULT 0,  
    wins INT DEFAULT 0,  
    losses INT DEFAULT 0,  
    points_scored INT DEFAULT 0,  
    points_conceded INT DEFAULT 0,  
    field_goal_percentage DECIMAL(5,2) DEFAULT 0.00,  
    three_point_percentage DECIMAL(5,2) DEFAULT 0.00,  
    FOREIGN KEY (leagueID) REFERENCES League(leagueID) ON DELETE CASCADE  
);
```

Figure 1.3: SQL Commands for table of creation of Team entity.

```

CREATE TABLE Player (
    playerID INT AUTO_INCREMENT PRIMARY KEY,
    teamID INT NOT NULL,
    playerName VARCHAR(100) NOT NULL,
    games INT DEFAULT 0,
    total_points INT DEFAULT 0,
    total_assists INT DEFAULT 0,
    total_rebounds INT DEFAULT 0,
    total_steals INT DEFAULT 0,
    total_blocks INT DEFAULT 0,
    total_turnovers INT DEFAULT 0,
    field_goal_percent DECIMAL(5,2) DEFAULT 0.00,
    FOREIGN KEY (teamID) REFERENCES Team(teamID) ON DELETE CASCADE
);

```

Figure 1.4: SQL Commands for table of creation of Player entity.

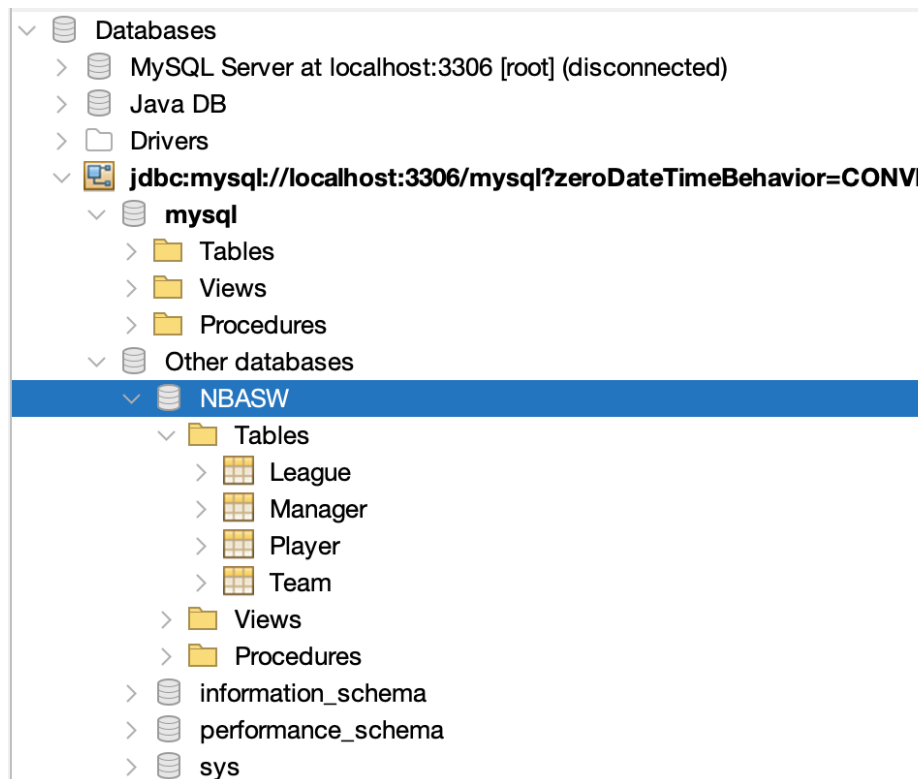


Figure 1.5: All created tables in the NBA stats website database.