

Project Proposal

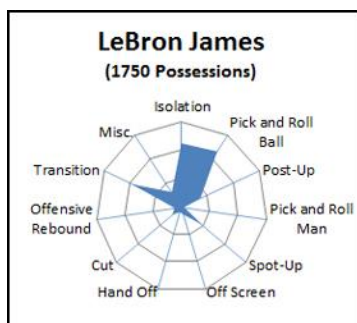
Yong Hao (yh3290), Jerry Liu (jl6007)

We are planning to implement a Python application that operates the NBA database through a simple web front end. The database is designed to serve NBA fans, scouts, data analysts, etc., to help them compare players and analyze games throughout multiple seasons. Users can achieve the statistics of all NBA players and teams. Records can also be inquired by sorting certain domains, such as points, rebounds, assists, etc. The whole design includes six entities: Season, Team, Player, Game, Career Stats, Arena. Career Stats is a weak entity. Details can be found in E-R diagram. The featured function of the application is to help users achieve data views in a more intuitive way – users can get the “radar chart” of a player, a team, or a coach that they inquiry. On top of that, users can choose specific perspectives to generate the radar chart they need and do the comparison.

We had a meeting with **Mavis Athene U Chen** on September 22nd at 11:15 am and she approved our project proposal. Then we made some modifications based on her advice.

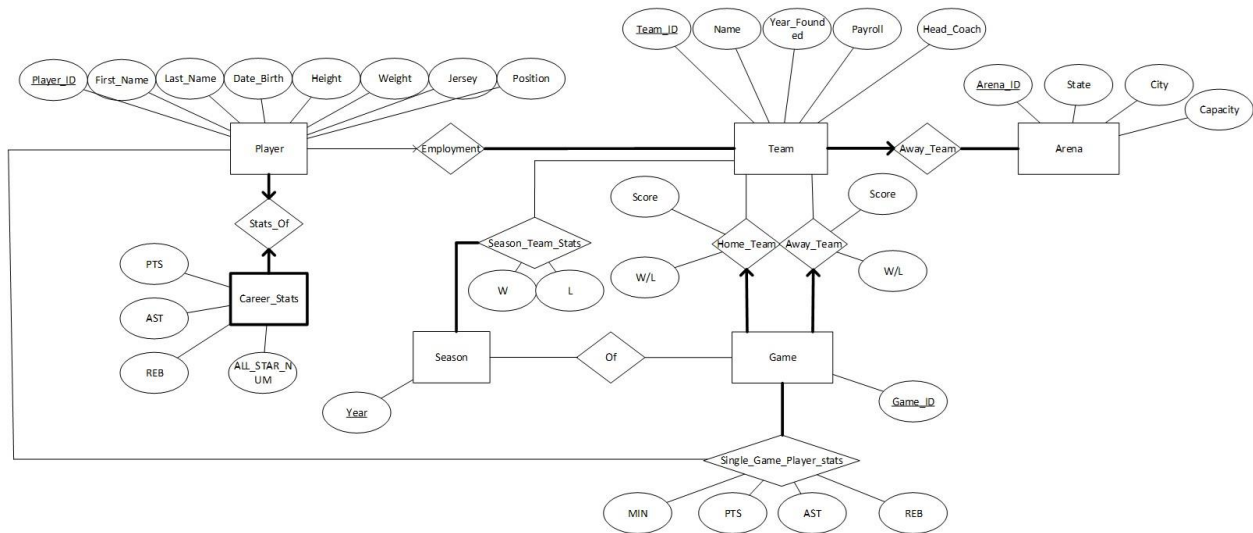
Part 3 Choice: A. Web Front-End Option

Radar Chart Sample:



Contingency Plan: If anyone of us dropped the class, the other one should finish the project but with fewer attributes per entity (for example, the entity *team* could delete *Year_Founded*, *Arena*, *City*, etc).

ER Diagram



Relational Schema

Player (Player_ID, First_Name, Last_Name, Date_Birth, Height, Weight, Jersey, Position, Team_ID,

PK(Player_ID), FK(Team_ID) -> Team)

Team (Team_ID, Name, Year_Founded, Payroll, Head_Coach,

PK(Team_ID))

Arena (Team_ID, Arena_ID NOT NULL, State, City, Capacity,

PK(Team_ID, Arena_ID), FK(Team_ID) -> Team)

Career_Stats (Player_ID, PTS, AST, REB, ALL_STAR_NUM,

FK(Player_ID) -> Player

ON DELETE CASCADE)

Season_Team_Stats (Year, Team ID, W, L,

PK(Year, Team_ID), FK(Team_ID) -> Team)

Game (Year, Game_ID, Home_Team NOT NULL, Home_Team_Score NOT NULL, Home_Team_W/L NOT

NULL, Away_Team NOT NULL, Away_Team_Score NOT NULL, Away_Team_W/L NOT NULL,

PK(Game_ID), FK(Home_Team, Away_Team) -> Team)

Single_Game_Player_Stats (Game_ID, Player_ID NOT NULL, MIN, PTS, AST, REB,

PK(Game_ID, Player_ID), FK(Game_ID) -> Game, FK(Player_ID) -> Player)

Note: For the ≥ 1 constraints. We will implement it after the class covers the part.