

Project Report

Football Database

Scenario and Database Requirements:

The football tournament database is a single comprehensive database that captures the information about various leagues and cup tournaments throughout the world. It also encapsulates the information about various football teams, their players and the staff associated with the clubs. The sponsor information of both the clubs and the leagues is also maintained in the database. Additionally, the database also manages to store the data points about match officials who officiate various leagues and cups. The combination of leagues and tournament in the same entity tournament allows the database to dynamically capture the insights of both types of football competitions allowing for a more streamlined structure.

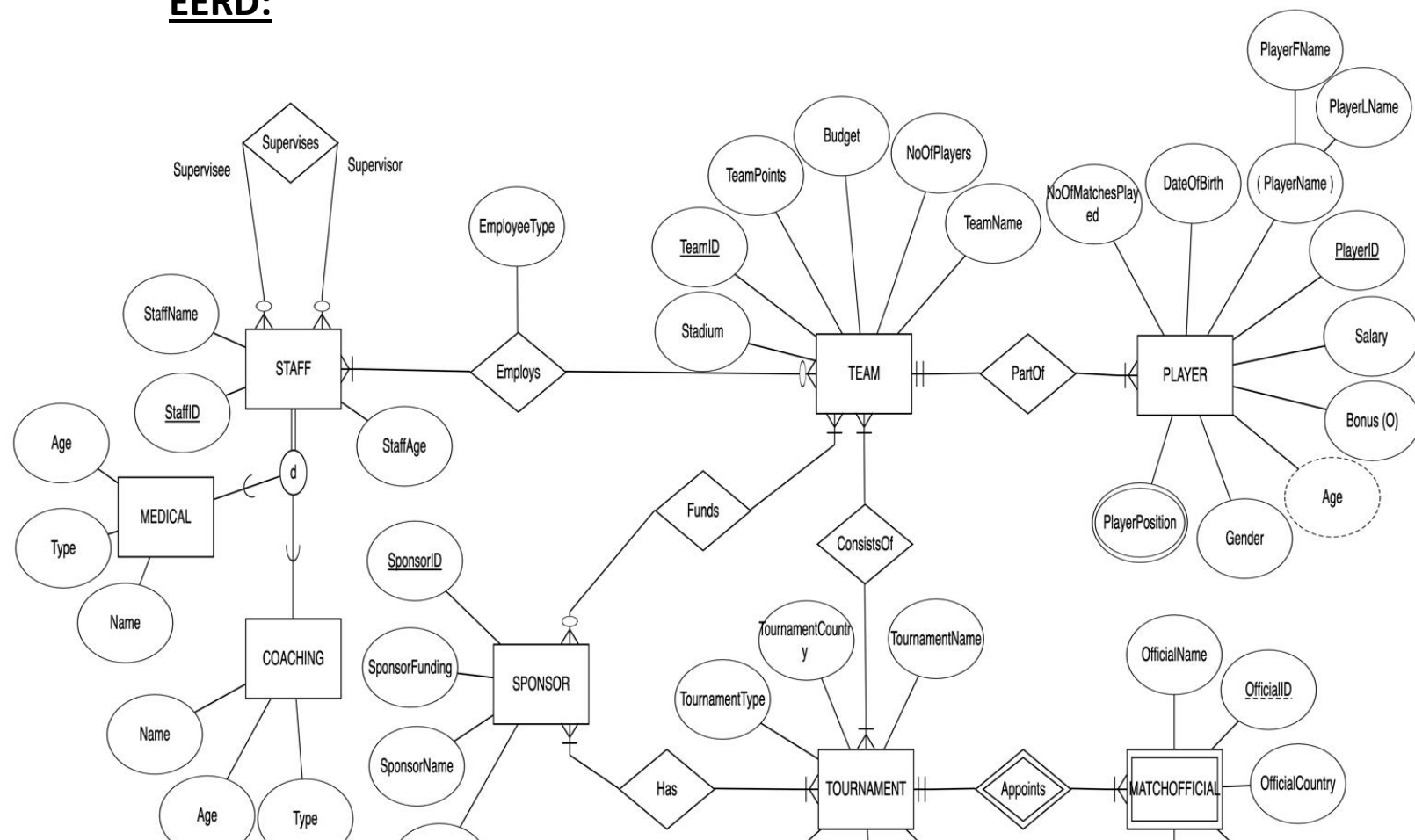
The main goal of this project is to create a multifaceted database that captures all the relevant information related to a football club.

The relation will keep track of football leagues, teams, sponsors, players, staff, and match official.

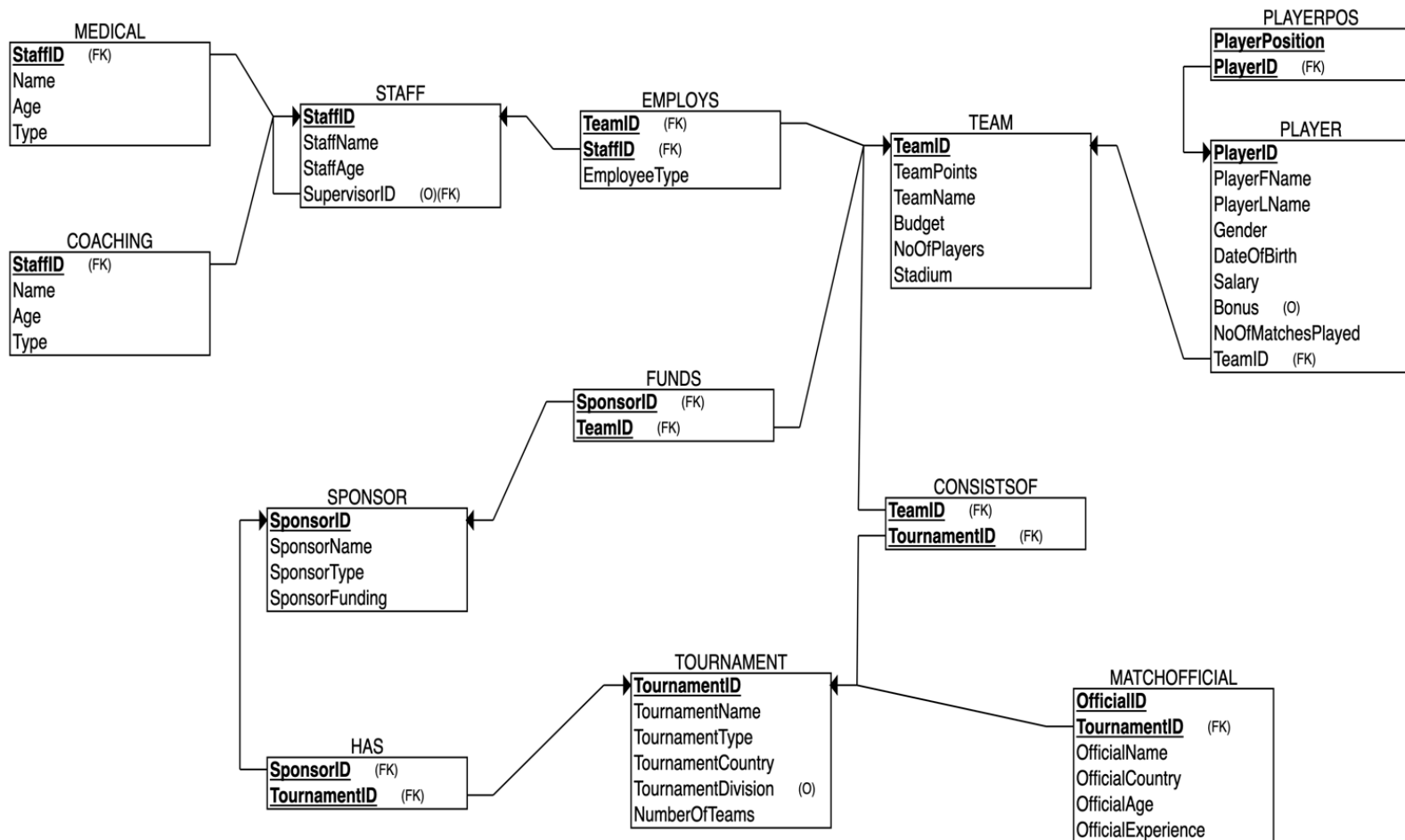
- For each team, there is a primary key team ID. It also keeps track of team name, stadium, team points, budget, and number of players.
- The player entity keeps track of player ID, player name (composite attribute), date of birth, gender, player position (multivalued attribute), age (derived attribute), salary, bonus (optional attribute) and number of matches played
- For each tournament ID (primary key), tournament name, tournament country, tournament type, tournament division (optional) and number of teams are stored in the database
- The staff entity keeps track of staff name, staff age and staff ID (primary key).
- Medical is a sub-entity of staff consisting of age, type, and name of the medical staff.

- Coaching is also a sub-entity of staff consisting of name, age, and type of the coaching staff.
- The match official entity keeps track of official name, official country, official age, official experience, and official ID (partial key)
- Each sponsor will have a sponsor ID, sponsor name, sponsor funding and sponsor type.
- A tournament has at least one sponsor.
- A team may have one sponsor.
- A tournament appoints at least one match official to officiate the tournament.
- A match official can be appointed to exactly one tournament.
- A team employs multiple staff members. This relationship is specified using the employee type attribute.
- A staff member may work for at least one or more teams.
- A player is a part of exactly one team.
- A team includes many players.
- Staff has a recursive relation with itself. Indicating the relationship between a supervisor and a supervisee.

EERD:



Relational Schema:



Normalization:

Almost all tables in the dataset are in third normal form. There is one exception, the table 'Player' is in second normal form. This is because the Salary column is transitively dependent on the NoOfMatchesPlayed column. However, it is unlikely

that many players have the same salary, therefore different contracts. So, no point making a new table.

Database Creation and Population:

Data Sources:

To create a database, we used a combination of publicly available datasets from Kaggle, Wikipedia and Data World. We have designed a data schema that requires us to use some auxiliary data. For example: the staff table required us to use our imagination as information about club staff is not publicly available in most cases. We used the following websites and datasets to fill in our database.

- <https://www.kaggle.com/datasets/slehkyi/extended-football-stats-for-european-leagues-xg>
- <https://www.kaggle.com/datasets/hugomathien/soccer>
- <https://data.world/datasets/soccer>
- <https://www.kaggle.com/datasets/ido92/epl-stats-20192020>
- https://en.wikipedia.org/wiki/La_Liga

Data Types:

For the **Player** entity:

We decided to fixed length CHAR of size 5 for the PlayerID. As PlayerID is a unique key assigned to every player. In which the first two characters are the country code of the player followed by a unique 3-digit number. For PlayerFName(25), PlayerLName(25), PlayerPosition(10) and Gender(10) we used variable length VARCHAR.

For Age (derived attribute) we used SMALLINT datatype with size 2. For Bonus (Which is optional) and fixed attribute Salary we used a normal size floating point number DOUBLE. For NumberOfMatches we used INT of size 4.

For the DateOfBirth attribute we used the DATE data type where the supported range is (1000-01-01 to 9999-12-31)

All the IDs in our database for the different entities like Team, Player, Tournament, etc. are unique and not NULL, and are acting as the primary keys for these entities. All these IDs are of the data type VARCHAR which are of variable length, except for the TeamID.

SQL Statements:

Query 1:

The following query can be used by the user to find out the players that play in certain teams that have a salary above the average salary and have played at least a certain number of matches. The purpose of this query is to see the optimal players that have a good number of matches played and are too expensive. Parameters of the query (like salary range) can be changed to fit different needs. The query mainly uses an inner join between the team and player tables so that a specific team can be targeted and the players with the given parameters be selected accordingly. Very specific columns are being selected from both tables to maximize the amount of useful information displayed to efficiently gain insights from the results.

Code:

```
SELECT DISTINCT p.PlayerFName, p.PlayerLName, p.DateOfBirth, p.Salary,
p.NoOfMatchesPlayed, t.TeamName
FROM Player p
INNER JOIN Team t
ON p.TeamId = t.TeamID
WHERE p.salary >
(SELECT AVG(salary) FROM player )
AND p.NoOfMatchesPlayed >= 45;
```

Result:

100% 31:248

Result Grid

Filter Rows:

Search

Export:

	PlayerFName	PlayerLName	DateOfBirth	Salary	NoOfMatchesPlayed	TeamName	
▶	Kevin	Debruyne	1991-06-25	300000	45	Manchester City	
	Cristiano	Ronaldo	1985-02-05	500000	50	Manchester United	


Query 2:






This query can be used to correctly identifying the 'premium' tournaments in the database. A 'premium' tournament can be defined as that which has at least 15 teams in that tournament and has a funding of at least \$500000 to begin with. The purpose can be further interpolated as those tournaments which have the greatest fan followings and ratings since they will be heavily advertised, hence the massive funding.






Code:

```
SELECT SponsorID, TournamentID
FROM Has
WHERE TournamentID IN (SELECT TournamentID
                        FROM Tournament
                        WHERE NumberOfTeams >= 15)
AND SponsorID IN (SELECT SPONSORID
                  FROM SPONSOR
                  WHERE SponsorFunding >= 500000);
```

Result:

100%  53:254

Result Grid   Filter Rows: Edit:    Export/Import

	SponsorID	TournamentID	
	S02	A2	
	S03	A3	
	NULL	NULL	
			
			


Query 3:

In the following query, we are selecting the player's name, the number of matches played by the player, the players position of all players who play in a team which has more than 20 total number of players. In addition, such teams should also employ an employee with type C, indicating a coach. This query can be used by fans to identify the players and their position who play in a team with a coach and a certain number of players. This allows the fans of certain football clubs or football teams to get specific insights.

Code:

```
SELECT p.playerFname, p.playerLname, p.Noofmatchesplayed, ps.playerposition
FROM player p
JOIN playerpos ps ON ps.playerid = p.playerid
WHERE EXISTS (SELECT 1
               FROM team t
               WHERE t.teamid = p.teamid
               AND t.noofplayers > 20
               AND t.teamid in
               (SELECT teamid
                FROM employs
                WHERE employeetype = 'C'))
LIMIT 1;
```

Result:

100%	↺	49:264			
Result Grid			Filter Rows:	Search	Export: 
	playerFname	playerLname	Noofmatchesplayed	playerposition	
▶	Martin	Odegaard	15	CAM	
▶					
▶					


Query 4:


The purpose of this query is to identify the top players in the Tournament, which is located in England, where the teams the players are playing for have more than 60 more points and are on the top half of the table. This will help other teams to scout top players and try and sign them, and accordingly recruit them. England is a country which has its own leagues and is very well known for football, hence a higher user database. This is the reason why England is targeted specifically.


Code:


```
SELECT p.PlayerFName, p.PlayerLName, p.Gender, te.TeamID, te.TeamName,
tou.TournamentID, tou.TournamentName
FROM team te
NATURAL JOIN tournament tou
NATURAL JOIN player p
WHERE te.TeamPoints >= 60
AND tou.TournamentCountry = "England"
ORDER BY te.TeamID;
```





Result:

100%  39:272

Result Grid 

 Filter Rows:

Export: 

	PlayerFName	PlayerLName	Gender	TeamID	TeamName	TournamentID	TournamentName	
	Kevin	Debruyne	Male	1	Manchester City	A3	Premier League	
	Mo	Salah	Male	3	Liverpool	A3	Premier League	
	Kai	Havertz	Male	4	Chelsea	A3	Premier League	
								

Further Discussion:

The database can be further expanded upon to include more leagues and teams outside of Europe. We can add additional entities in the future so that end users can gain more insights by running more intricate queries. We feel that our current built Football Tournament database is a good starting point to eventually encapsulate all data points of modern football.