DBMS Project Report

PES University

Database Management Systems

UE18CS252

Submitted By

PES2201800508 NAVEEN SURESH

This Project is based on a football management system, where users can access information about the players, various football clubs, football managers, team information etc. from the database. There are two types of accounts in the system, namely the regular accounts and the premium accounts. Regular accounts have access only to the players' basic details, whereas premium accounts have access to deeper information on the players, managers and the clubs.

The entities consists of information corresponding to the players, managers and clubs. They also contain data about the activity of the users. Triggers were implemented based on identifying constraints related to premium account users.

Various SQL queries that perform a variety of operations have been implemented to enrich the user experience by providing profound information on the preferred search.

Introduction

This Project is based on a football management system, where users can access information about the players, various football clubs, football managers, team information etc. from the database. The Users can create an account in order to experience the various features of the software. There are two types of accounts in the system, namely the regular accounts and the premium accounts. Regular accounts can access only details about the players and nothing else. The regular account users, can enter their preferred player name they want to search for. The details of the corresponding player will be retrieved from the ‘all players’ table and displayed. This is the only feature that the regular account holders can experience.

The premium account holders have access to wide range of details, statistics and information about the clubs, players and managers. For premium users there is an option in the home screen navigation bar called as premium, where they will be asked to enter their account credentials. Apart from the players’ basic details, the

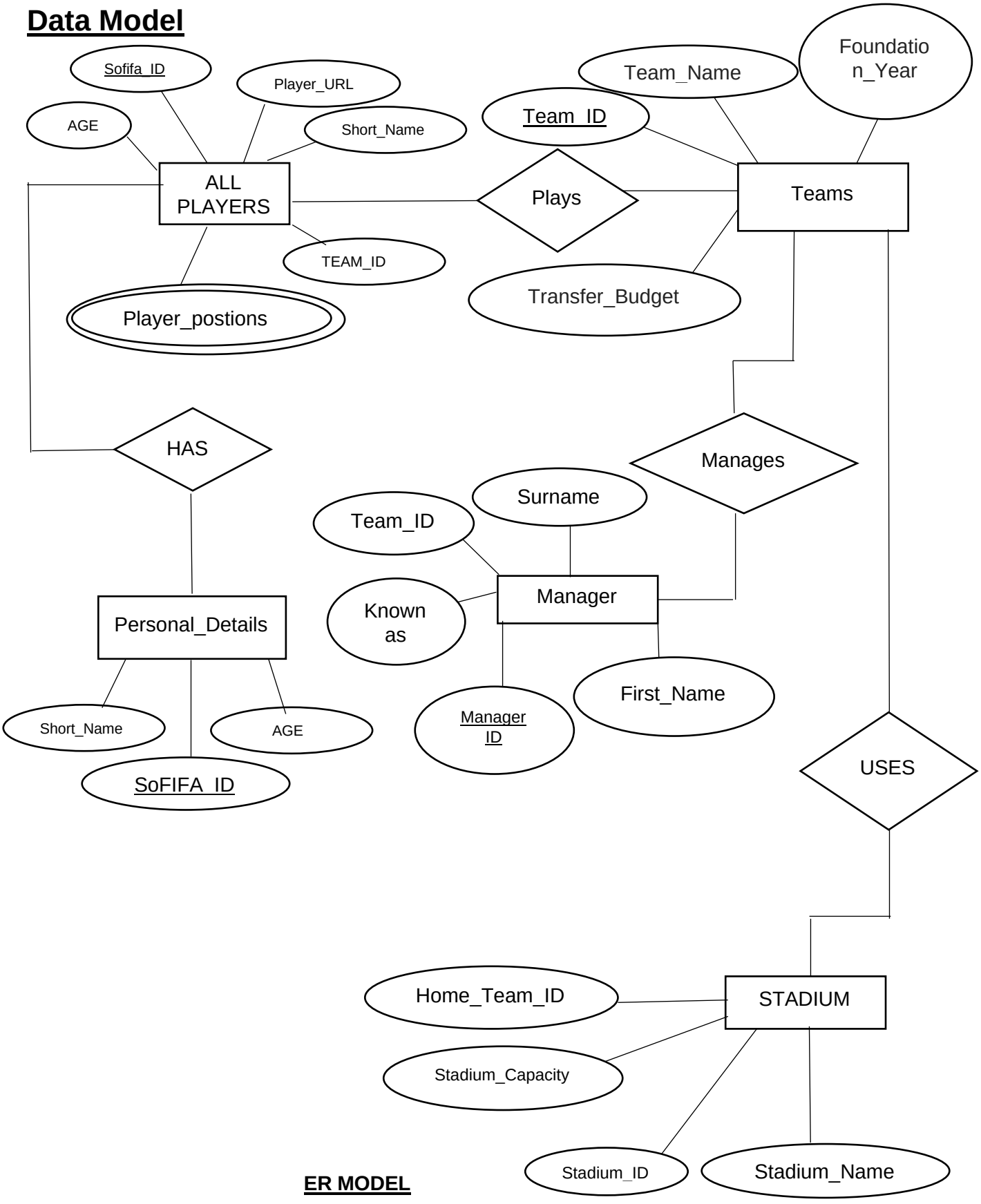
premium users have access to height, weight, weekly wage, jersey number, preferred playing position etc. The premium users also have access to plenty of information about the clubs.

Details about the entities:

1) **The All players** – This table consists of the data of all the players from various clubs, with attributes/columns such as height, name etc. The Primary key for this table is the SOFIFA_ID which is the Player ID that is unique to every player. The foreign key is Team_ID.
2) **The Login_track** – This table is used for keeping a track of the users who have logged in. It also helps in storing the details of the users' log in time for their current session as well as previous session. The attributes of this table include the username, cur_time(which keeps of track of the current session's log in time, prev_time(which keeps track of the previous session's log in time of a user), count, points, streak. The Primary Key as well as the foreign key(referenced from users) for this table which is the id (this to be seen again).
3) **The Manager** – This table consists of information on the managers belonging to various football clubs. The Primary key here is the manger id, which is unique to every manager. The Attributes of this table include First name, surname, Head Asset ID, Team_ID. The foreign key(referenced from here is the Team_ID.
4) **The Personal Details** – Primary key and foreign key(referenced from All players table) – sofifa_id. User can edit only those aspects that were mentioned in the table except for primary key. The changes will be reflected in this table as well as all players table.

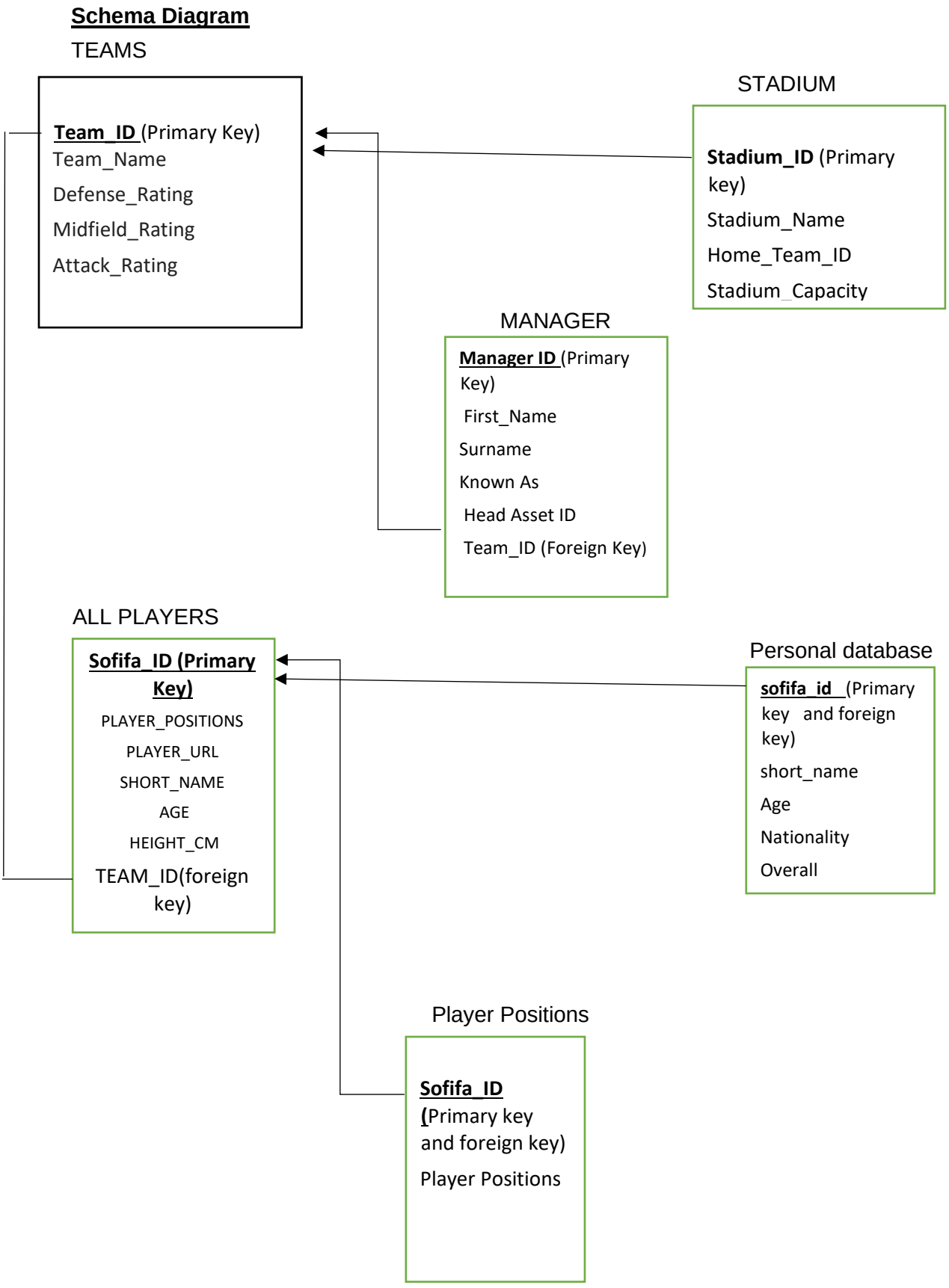
5) **The Premium_logs** – This table records if the user has been updated to premium using triggers. The attributes of this table are ID of the user, the name of the user and the time of login.
6) **The Stadiums** –This table contains the information on various stadiums. It has attributes such as Stadium_name, Stadium_id, Stadium_capacity, Hometeam_id(which corresponds to the ID of the team for which the stadium is the home ground). The primary key here is the Stadium_id and the foreign key is the Homteam_id.
7) **The Update_logs** – This table's records get updated whenever admin updates user data. The attributes include, time which signifies the updated time, action which explains the action undertaken by the admin, and the ID, which is the id of the user whose data has been updated.
8) **The Users** – This table contains data of all the users. The attributes are id, username, email, user_type(which signifies admin or normal user), password and premium(which tells whether the user is a premium user or not). The primary key is id.
9) **The Teams** – The teams table contains information about the teams, such as Team_ID, Team_Name, Defense_Rating, Midfield_Rating, Attack_Rating, Overall_Rating, Transfer_Budget, Foundation_Year, League_Titles. Here, the primary key is Team_ID.

Data Model



ER MODEL

Schema Diagram



FD and Normalization

1) **All Players** (SOFIFA_ID, PLAYER_URL, SHORT_NAME, AGE, HEIGHT_CM, WEIGHT_KG, NATIONALITY, CLUB, OVERALL, POTENTIAL, VALUE_EUR,WAGE_EUR, PLAYER_POSITIONS, PREFERRED_FOOT, WEAK_FOOT,SKILL_MOVES,RELEASE_CLAUSE_EUR,TEAM_POSITION,TEAM_JERSEY_NUMBER,PACE,SHOOTING,PASSING,DRIBBLING,DEFENDING, PHYSIC, TEAM_ID)

   FD = {SOFIFA_ID → PLAYER_URL, SHORT_NAME, AGE, HEIGHT_CM, WEIGHT_KG, NATIONALITY, CLUB, OVERALL, POTENTIAL, VALUE_EUR,WAGE_EUR, PLAYER_POSITIONS, PREFERRED_FOOT, WEAK_FOOT,SKILL_MOVES,RELEASE_CLAUSE_EUR,TEAM_POSITION,TEAM_JERSEY_NUMBER,PACE,SHOOTING,PASSING,DRIBBLING,DEFENDING, PHYSIC, TEAM_ID}

2) **Login_track**(id, action, username, cur_time, prev_time, count, points, streak)

   FD = {id->action,username,cur_time,prev_time,count, points, streak }

3) **Manager**(Manager ID, First_Name, Surname, Known As, Head Asset ID, Team_ID)

   FD = {Manager ID -> First_Name, Surname, Known As, Head Asset ID, Team ID}

4) **Personal_details**(sofifa_id, short_name, age, nationality, overall)

   FD = {sofifa_id -> short_name, age, nationality, overall}

5) **Premium_logs**{id, name, action, time}

   FD = {id -> name, action, time}

6) **Stadiums**(Stadium_ID, Stadium_Name, Home_Team_ID, Stadium_Capacity)

   FD = {Stadium_ID -> Stadium_Name, Home_Team_ID, Stadium_Capacity}

7) **Users** (id,username,email,user_type,password,premium)

   FD = {id -> username,email,user_type,password,premium}

8) **Update_logs**(id,action,time)

   FD = {id->action,time}

9) **Teams**(Team_ID,Team_Name,Defense_Rating,Midfield_Rating,Attack_Rating, Overall_Rating, Transfer_Budget,Foundation_Year,League_Titles)

FD = {Team_ID- > Team_Name, Defense_Rating, Midfield_Rating, Attack_Rating, Overall_Rating, Transfer_Budget,Foundation_Year,League_Titles}

Normalisation

Most tables and relations have been normalised at least to the 3nf to ensure there is not much data redundancy. But, in a few tables this is not possible due to some missing data as this would decrease the efficiency while performing the queries and joins.

For example, if we combine the Defensive_Rating, Midfield_Rating, Attack_Rating in the Teams table, then it will even violate the 1nf as we have non atomic column.

DDL

1) All Players

```
CREATE TABLE `allplayers` (`SOFIFA_ID` int(6) NOT NULL, `PLAYER_URL` varchar(78) DEFAULT NULL, `SHORT_NAME` varchar(22) DEFAULT NULL, `AGE` int(2) DEFAULT NULL,`HEIGHT_CM` int(3) DEFAULT NULL, `WEIGHT_KG` int(3) DEFAULT NULL, `NATIONALITY` varchar(20) DEFAULT NULL,`CLUB` varchar(35) DEFAULT NULL, `OVERALL` int(2) DEFAULT NULL, `POTENTIAL` int(2) DEFAULT NULL, `VALUE_EUR` int(9) DEFAULT NULL, `WAGE_EUR` int(6) DEFAULT NULL, `PLAYER_POSITIONS` varchar(12)
```

DEFAULT NULL, `PREFERRED_FOOT` varchar(5) DEFAULT NULL, `WEAK_FOOT` int(1) DEFAULT NULL,`SKILL_MOVES` int(1) DEFAULT NULL,`RELEASE_CLAUSE_EUR` varchar(9) DEFAULT NULL,`TEAM_POSITION` varchar(3) DEFAULT NULL,`TEAM_JERSEY_NUMBER` varchar(2) DEFAULT NULL, `PACE` varchar(2) DEFAULT NULL, `SHOOTING` varchar(2) DEFAULT NULL,`PASSING` varchar(2) DEFAULT NULL, `DRIBBLING` varchar(2) DEFAULT NULL, `DEFENDING` varchar(2) DEFAULT NULL, `PHYSIC` varchar(2) DEFAULT NULL, 'TEAM_ID' int(5) DEFAULT NULL)

2) Login_track

CREATE TABLE `login_track` (`id` int(10) NOT NULL,`action` varchar(10) NOT NULL, `username` varchar(20) NOT NULL,`cur_time` timestamp NOT NULL DEFAULT current_timestamp(),`prev_time` timestamp NOT NULL DEFAULT current_timestamp(),`count` int(10) NOT NULL,`points` int(10) NOT NULL,`streak` int(10) NOT NULL)

3) Personal_details

CREATE TABLE `personal_details` (`sofifa_id` int(20) NOT NULL,`short_name` varchar(50) NOT NULL,`age` int(20) NOT NULL,`nationality` varchar(50) NOT NULL,`overall` int(10) NOT NULL)

4) Premium_logs

```
CREATE TABLE `premium_logs` (`id` int(10) NOT NULL,`name` varchar(20) NOT NULL,`action` varchar(50) NOT NULL,`time` timestamp NOT NULL DEFAULT current_timestamp())
```

5) Stadiums

```
CREATE TABLE `stadiums` (`Stadium_ID` int(4) NOT NULL,`Stadium_Name` varchar(42) DEFAULT NULL,`Home_Team_ID` int(6) DEFAULT NULL,`Stadium_Capacity` int(5) DEFAULT NULL)
```

6) Teams

```
CREATE TABLE `teams` (`Team_ID` int(6) NOT NULL,`Team_Name` varchar(28) DEFAULT NULL,`Defense_Rating` varchar(2) DEFAULT NULL,`Midfield_Rating` int(2) DEFAULT NULL,`Attack_Rating` int(2) DEFAULT NULL,`Overall_Rating` int(2) DEFAULT NULL,`Transfer_Budget` int(9) DEFAULT NULL,`Foundation_Year` int(4) DEFAULT NULL,`League_Titles` int(2) DEFAULT NULL)
```

7) Update_logs

CREATE TABLE `update_logs` (`id` int(10) NOT NULL,`action` varchar(50) NOT NULL,`time` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE current_timestamp())

Triggers

The constraints identified in order to implement triggers are:

1) To keep track of users who upgraded to premium.

```
BEGIN
    IF OLD.premium like 'no' and new.premium like 'yes' THEN
        INSERT INTO premium_logs(id,name,action,time)
        VALUES(old.id,old.username,'User with upgraded to Premium',NOW());
    END IF;
END
```

2) To keep track of the players whose data has been updated.

```
INSERT INTO update_logs(action,time) VALUES ("Updated the values in allplayers table",NOW())
```

SQL Queries

To calculate the value of a football club by taking adding up each player's value if he plays for the club.

```
SELECT SUM(value_eur) from allplayers where club  LIKE \"$input_name\
```

To calculate the average rating of the Club.

SELECT AVG(Overall) from allplayers where club LIKE \"$input_name\

To select all the attributes from login_track and users table for the user who has currently logged in.

SELECT login_track.cur_time,login_track.prev_time,login_track.id,login_track.username,users.premium from login_track inner join users ON users.id=login_track.id where users.username='$username'";

To select the top 5 highest rated players from evey club from the all players table.

SELECT short_name,Overall from allplayers where club LIKE \"$input_name\" order by Overall DESC Limit 5";

To deduct 50 points from the user who has upgraded to premium.

UPDATE login_track SET points=points-50 where username='$username'

To return the details of a manager for the searched club from inner join of manager and teams.

SELECT manager.First_Name,manager.Surname from manager inner join teams ON teams.Team_ID=manager.Team_ID WHERE Team_Name LIKE \"$input_name\"

To update the details of the personal_details table whenever the admin updates the details of the user.

UPDATE personal_details SET short_name ='" . $input['short_name'] . "',age ='" . $input['age'] . "',overall ='" . $input['overall'] . "', nationality='" . $input['nationality'] . "'" ." WHERE sofifa_id='" . $input['sofifa_id'] . "'"

Conclusion

The football management system that has been developed in this project allows users to access information related to football players, clubs and other aspects related to the game. The system acts as a one point destination for users eager to know about the various details and statistics of the game. It allows users to create accounts and rewards them with points based on their activity in the system and upgrades them to premium if their activity matches a certain criteria. As a premium account holder, the user gets exposed to profound information about various clubs, managers and the skill set of the players. One of the limitations is that we cannot delete player data. Another limitation the admin cannot manage the users. The future enhancements can have the features of football news, fixtures and the up gradation to premium account can be based on subscription basis.