

# ITUDB2320 - PROJECT REPORT FOR GOALALCHEMY

Ege Demir, 150200319  
Zehra Demir, 150200305  
Havva Eda Körpe, 150200029

03/01/2024

## 1 Technology

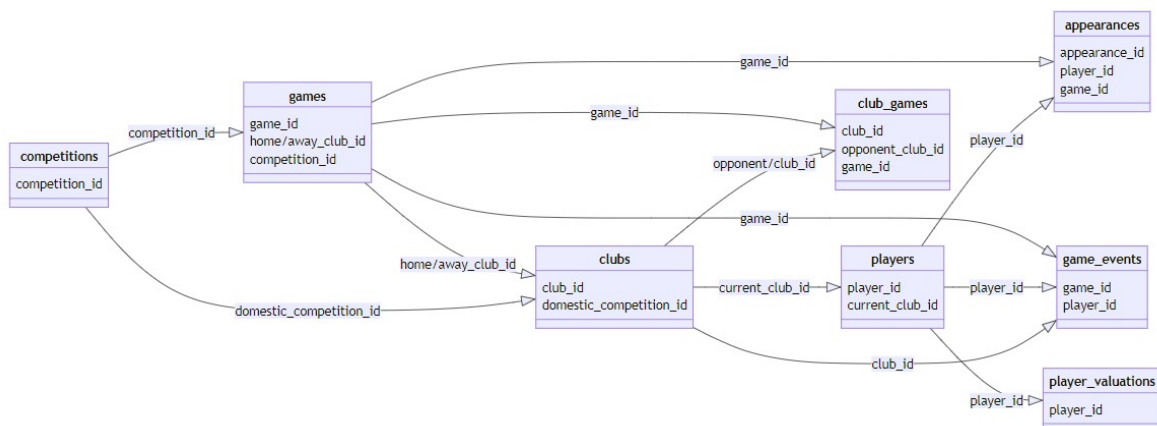
Throughout this project, Python programming language has been used with MySQL as the Relational DataBase Management System tool. Flask web framework has been used for web application, and for frontend, HTML has been utilized with CSS for styling and JavaScript for some scripts. Additionally, the pandas library of Python programming language has been used for data cleaning.

## 2 Dataset

For this project, we used football data acquired from Transfermarkt. Dataset can be reached from here: <https://data.world/dcereijo/player-scores>

### 2.1 Keys

The diagram to show the primary keys, foreign keys, and the tables that these foreign keys reference to can be seen below. 6 out of 8 shown tables have been used as main tables. Player valuations and game events tables have been discarded because of unnecessary or redundant data they possess.



## **2.2 Responsibility Distribution**

### **2.2.1 Ege Demir - 150200319**

Responsible for 'games table' and 'clubs' table.

'games' table will be used for game results tracing system. Shows club names, competition, score, date season etc. Users must be able to filter by club name, competition, season, date; and sort by goal difference.

'clubs' table will be used for club features tracing system. Shows club code, name, player statistics etc. Users must be able to filter by club code, name, competition; and sort by squad size, average age, foreigners number, foreigners percentage, national team players.

### **2.2.2 Zehra Demir - 150200305**

Responsible for 'appearances' table and 'players' table.

'appearances' table will be used for player performance tracing system. Shows goals, assists, yellow cards, opponent club etc. Users must be able to filter by player name, club name, season; and sort by goal count, maximum game count.

'players' table will be used for player features tracing system. Shows player name, last season, club, place of birth, citizenship. Users must be able to filter by name, player code, place of birth, citizenship; and sort by last season.

### **2.2.3 Havva Eda Körpe - 150200029**

Responsible for 'club games' table and 'competitions' table

'club games' table will be used for club situation and performance tracing system. Shows the players, manager, recent results of teams. Users must be able to filter by manager name, club name, hosting; and sort by positions and goal counts in descending order.

'competitions' table will be used for competition information tracing system. Shows name, type, subtype, country name, etc. Users must be able to filter by competition code, name, type, and country; and sort by them in the ascending order.

## **2.3 Database Configuration Details**

After installing MySQL with 'root' as our username and 'localhost' as the host, we all created a database named 'football' in MySQL. With these installation details, 3 out of 4 items to connection were the same for all 3 of us with the exception of password. So we had to take input for password in config.py file shown below. With this way, we are all able to reach our database

and work on it by running the same python scripts.

```
config.py > ...  
1 db_host = 'localhost'  
2 db_user = 'root'  
3 db_password = str(input("Enter your admin password: "))  
4 db_database = 'football'
```

Figure 1: config.py File

## 3 Website Details

### 3.1 Home Page

On the home page, this menu welcomes us with 6 buttons that take us to the page of each table.

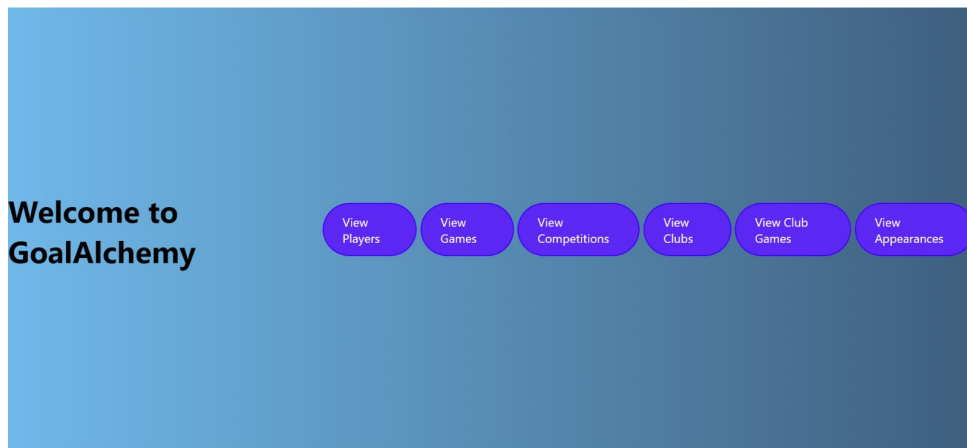


Figure 2: Home Page of the Website

### 3.2 View Table Page

After clicking any of the buttons in the home menu, we get to the 'view table' menu of the corresponding table. From here, it's possible to update or delete any existing entries. Update and delete buttons at the rightmost columns in each row take the user to the update/delete page. At the upper-left side of the page, it's possible to go to the 'sort/filter' or 'add record' pages that we will mention in later sections (Fig. 3).

Sort or Filter Games

Add Game

Football Games













Game ID	Competition Name	Home Club Name	Away Club Name	Score	Season	Round	Date	Home Club Manager Name	Away Club Manager Name	Stadium	Attendance	Referee Name	Update	Delete
2211607	johan cruijff schaal	PSV Eindhoven	Ajax Amsterdam	4-2	2012	Final	2012-08-05	Dick Advocaat	Frank de Boer	Johan Cruijff ArenA	50000	Björn Kuipers		
2218677	supercoppa italiana	Juventus FC	SSC Napoli	4-2	2012	Final	2012-08-11	Massimo Carrera	Walter Mazzarri	National Stadium (Bird's Nest)	60000	Paolo Silvio Mazzoleni		
2219794	belgian supercup	RSC Anderlecht	ksc lokeren	3-2	2012	Final	2012-07-22	John van den Brom	Peter Maes	Constant Vanden Stock Stadion	14485	Laurent Colomonts		
2219795	uefa super cup	Chelsea FC	Atlético de Madrid	1-4	2012	Final	2012-08-31	Roberto Di Matteo	Diego Simeone	Stade Louis-II	14312	Damir Skomina		
2221757	dfb pokal	arminia bielefeld	sc paderborn 07	3-1	2012	First Round	2012-08-19	Stefan Krämer	Stephan Schmidt	SchücoArena	18587	Markus Wingenbach		
2222360	europa league qualification	eskisehirspor	St. Johnstone FC	2-0	2012	Second Round 1st leg	2012-07-19	Ersun Yanal	Steve Lomas	Eskişehir Atatürk	12675	Ante Vucemilovic		

Figure 3: Games Page

And at the bottom of the page, it's possible to visit any page to see other records. Also by clicking the huge button at the bottom, it's possible to went back to main page. This attribute is available for every page (Fig. 4).

1	2	3	4	5	Next
Back to Main Page					

Figure 4: Pagination

Also in players page, there is a search bar which is used to search for players using either their full ID's or part of their names. It is because the query consists of "LIKE" for name and equal sign for ID (Fig. 5).

Search by name or player ID.

Sort and Filter PlayersCheck Player FeaturesAdd Player

Players

Player ID	First Name	Last Name	Full Name	Last Season	Current Club Name	Player Code	Country of Birth	City of Birth	Country of Citizenship	Date of Birth	Sub Position	Position	Foot	Height in cm	Market Value in EUR
10	Miroslav	Klose	Miroslav Klose	2015	SS Lazio	miroslav-klose	Poland	Opole	Germany	1978-06-09	Centre-Forward	Attack	right	184	None
26	Roman	Weidenfeller	Roman Weidenfeller	2017	Borussia Dortmund	roman-weidenfeller	Germany	Diez	Germany	1980-08-06	Goalkeeper	Goalkeeper	left	190	None
65	Dimitar	Berbatov	Dimitar Berbatov	2015	PAOK Thessaloniki	dimitar-berbatov	Bulgaria	Blagoevgrad	Bulgaria	1981-01-30	Centre-Forward	Attack	None	None	None
77	None	Lúcio	Lúcio	2012	Juventus FC	lucio	Brazil	Brasília	Brazil	1978-05-08	Centre-Back	Defender	None	None	None
80	Tom	Starke	Tom Starke	2017	Bayern Munich	tom-starke	East Germany (GDR)	Freital	Germany	1981-03-18	Goalkeeper	Goalkeeper	right	194	None
109	None	Dedé	Dedé	2012	None	dede	Brazil	Belo Horizonte	Brazil	1978-04-18	Left-Back	Defender	None	None	None

Figure 5: Search Bar in View Page

### 3.3 Add Record Page

After clicking the add record button, we see a page where it is possible to specify all columns for a new record. However, not all columns have to be filled to add a new record. The mandatory columns have been illustrated with red stars as it can be seen below. Also, it's possible to go

back to the view table page by clicking on the upper-left button.

The screenshot shows a web interface for adding a new game record. On the left, there is a blue button labeled "Back to Games". To the right is a white form titled "Add Game". The form contains the following fields: "Game ID:" with a value of "e.g. 1234", "Competition ID:" with a value of "e.g. ESI", "Season:" with a value of "e.g. 2020", "Round:" with a value of "e.g. 1", "Date:" with a value of "e.g. aa.yyyy" and a calendar icon, "Home Club ID:" with a value of "e.g. 100", "Away Club ID:" with a value of "e.g. 101", "Home Club Goals:" with a value of "e.g. 2", "Away Club Goals:" with a value of "e.g. 1", "Home Club Position:" with a value of "e.g. 3", and "Away Club Position:" with a value of "e.g. 5".

Figure 6: Inserting New Record

### 3.4 Update Page

After clicking the purple update button at the right side of each table page, an update page opens and the columns of the table can be updated using this page. In players' update page, there exist error handling which does not allow the user to input existing player id or invalid values such as entering different type to a value etc.

The screenshot shows a web interface for updating a player record. On the left, there is a blue button labeled "Back to Players". To the right is a white form titled "Update Player". The form contains the following fields: "First Name:" with a value of "Miroslav", "Last Name:" with a value of "Klose", "Last Season:" with a value of "Miroslav Klose", "Current Club ID:" with a value of "2015", "Player Code:" with a value of "398", "Country of Birth:" with a value of "miroslav-klose", "City of Birth:" with a value of "Poland", and "Country of Citizenship:" with a value of "Opole".

Figure 7: Updating the Record

### 3.5 Delete Page

After clicking the red delete button at the rightmost side of each table page, a message comes into the upper side of the page.

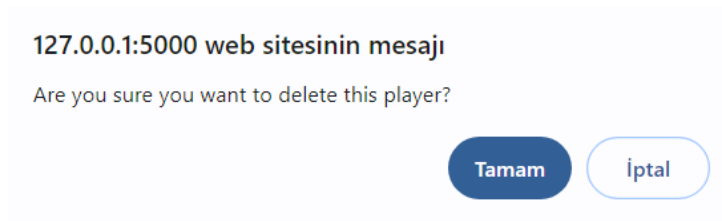


Figure 8: Website Delete Message

It is handled with the following JavaScript code inside the HTML file of the view\_player:

```
<script>
    function confirmDelete(playerId) {
        if (confirm("Are you sure you want to delete this player?")) {
            window.location.href = "/delete_player/" + playerId;
        }
    }
</script>
```

Figure 9: Javascript Code for Confirmation

After approving the message, another delete page opens for the final decision.

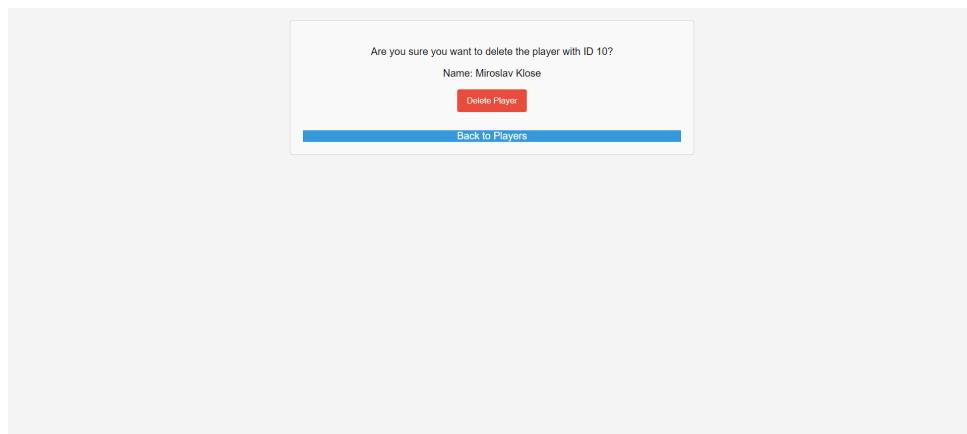


Figure 10: Deletion Page

By clicking delete player, player can be deleted from the records.

### 3.6 Sort/Filter Record Page

After clicking the sort/filter button, we get a menu where we can specify any attribute we want to search/filter by. We can enter a part of the string for any possible filters. After clicking 'apply filters' button or pressing the enter key, this page returns all suitable records. Also, there's a drop-down menu to choose which column the user wants to sort the records based on.

Games

Back to Games

Competition Name:  Round:  Home Club Manager Name:  Away Club Manager Name:

Home Club Name:  Away Club Name:  Aggregate:  Competition Type: 

Select Type

 Referee:

Sort by: 

Season

Attendance

Home Club Goals

Away Club Goals

Date

Apply Filters


Game ID		Home Club Name	Away Club Name	Score	Season	Round	Date	Home Club Manager Name	Away Club Manager Name	Stadium	Attendance	Referee Name
2211607	johan cruijff schaal	PSV Eindhoven	Ajax Amsterdam	4-2	2012	Final	2012-08-05	Dick Advocaat	Frank de Boer	Johan Cruijff ArenA	50000	Björn Kuipers
2218677	supercoppa italiana	Juventus FC	SSC Napoli	4-2	2012	Final	2012-08-11	Massimo Carrera	Walter Mazzarri	National Stadium (Bird's Nest)	60000	Paolo Silvio Mazzoleni
2219794	belgian supercup	RSC Anderlecht	ksc lokeren	3-2	2012	Final	2012-07-22	John van den Brom	Peter Maes	Constant Vanden Stock Stadion	14485	Laurent Colemonts
2219795	uefa super cup	Chelsea FC	Atlético de Madrid	1-4	2012	Final	2012-08-31	Roberto Di Matteo	Diego Simeone	Stade Louis-II	14312	Damir Skomina
2221757	dfb pokal	arminia bielefeld	sc paderborn 07	3-1	2012	First Round	2012-08-19	Stefan Krämer	Stephan Schmidt	SchücoArena	18587	Markus Wingenbach

Figure 11: Sort and Filter Page

### 3.7 Player Page

In addition, after clicking the links of the player\_ids in the players page, there opens up a page which shows players image with some information.

### Player Details



Player ID: 10  
 First Name: Miroslav  
 Last Name: Klose  
 Last Season: 2015  
 Current Club Name: SS Lazio  
 Player Code: miroslav-klose  
 Country of Birth: Poland

[Go Back to Players](#)

Figure 12: Player Details

## 4 Operations on Tables

### 4.1 Clubs Table

```
query = """SELECT c.*,
    REPLACE(comp.name, '-', ' ') AS competition_name
FROM clubs c
LEFT JOIN competitions comp
ON c.domestic_competition_id = comp.domestic_league_code
WHERE comp.type LIKE 'domestic_league'
ORDER BY club_id ASC
LIMIT %s OFFSET %s"""
```

Figure 13: Complex Query for Clubs

In the query above, we join clubs table with competitions table so we can use competition name instead of competition ID. Also its important to only join them when competition type is domestic league to prevent multiple clubs, and update competition name for clean result.

### 4.2 Games Table

```
query = """SELECT g.*,
    REPLACE(comp.name, '-', ' ') AS competition_name,
    CONCAT(CAST(g.home_club_goals AS CHAR), '-', CAST(g.away_club_goals AS CHAR)) AS score,
    home_club.name AS home_club_name,
    away_club.name AS away_club_name
FROM games g
LEFT JOIN competitions comp ON g.competition_id = comp.competition_id
LEFT JOIN clubs home_club ON g.home_club_id = home_club.club_id
LEFT JOIN clubs away_club ON g.away_club_id = away_club.club_id
ORDER BY g.game_id ASC
LIMIT %s OFFSET %s
"""
```

Figure 14: Complex Query for Games

In the query above, we join games table with competitions table, and clubs so we can use competition name instead of competition ID and club names instead of club IDs. Also we merge home club goals and away club goals as 'score' column.

### 4.3 Players Table

In players table, there also exists a 'Check Player Features' button to navigate through player features page. By clicking that button, following page opens:



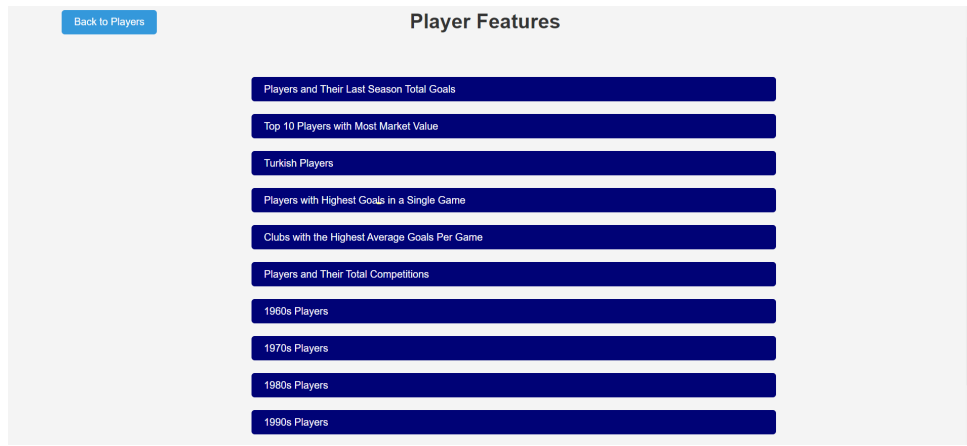


Figure 15: Player Features Page

There are complex queries and their results in the buttons. Some examples are shown below.

Turkish Players Page:

Turkish Players						
Player ID	Player Name	Current Club Name	Date of Birth	City of Birth	Country of Birth	Country of Citizenship
<a href="#">755</a>	Deniz Baris	Antalyaspor	1977-07-02	Erzincan	Turkey	Turkey
<a href="#">933</a>	Ugur Inceman	Konyaspor	1981-05-25	Aachen	Germany	Turkey
<a href="#">1791</a>	Halil Altintop	FC Augsburg	1982-12-08	Gelsenkirchen	Germany	Turkey
<a href="#">1792</a>	Hamit Altintop	None	1982-12-08	Gelsenkirchen	Germany	Turkey
<a href="#">1841</a>	Ali Bilgin	Kasimpasa	1981-12-17	Essen	Germany	Turkey
<a href="#">1944</a>	Deniz Dogan	None	None	Lübeck	Germany	Turkey
<a href="#">2453</a>	Mahir Saglik	None	1983-01-18	Paderborn	Germany	Turkey
<a href="#">2597</a>	Elyasa Süme	None	1983-08-13	Wesel	Germany	Turkey
<a href="#">5782</a>	Emre Belözoğlu	Fenerbahçe	1980-09-07	Istanbul	Turkey	Turkey
<a href="#">6800</a>	Orhan Ak	None	1979-09-29	Adapazarı	Turkey	Turkey

Figure 16: Turkish Players List

Players and Their Total Competitions Page:

Players and Their Total Competitions		
Player ID	Name	Competitions
<a href="#">238407</a>	Alexander Serloth	10
<a href="#">240306</a>	Bruno Fernandes	9
<a href="#">86202</a>	Antonio Rüdiger	9
<a href="#">40687</a>	Mathieu Valbuena	8
<a href="#">182932</a>	Davy Klaassen	8
<a href="#">172181</a>	Diogo Figueiras	8
<a href="#">55735</a>	Henrikh Mkhitaryan	8
<a href="#">124419</a>	Odysseas Vlachodimos	8
<a href="#">112515</a>	Marcos Alonso	8
<a href="#">16306</a>	Casemiro	8

1 2 3 4 5 Next

Back to Main Page

Figure 17: Players' Total Competition List

Below you can see a query example, which is for total goals of the players page.

```
query = """SELECT p.player_id, p.name, SUM(a.goals) AS total_goals
FROM players p
LEFT JOIN appearances a ON p.player_id = a.player_id
GROUP BY p.player_id, p.name
ORDER BY total_goals DESC
LIMIT %s OFFSET %s"""
```

Figure 18: Complex Query for Players

#### 4.4 Appearances Table

```
query = """SELECT c.club_id, c.name, AVG(a.goals) AS avg_goals_per_game
FROM clubs c
JOIN appearances a ON c.club_id = a.player Club_id
GROUP BY c.club_id, c.name
HAVING AVG(a.goals) > 0.15
ORDER BY avg_goals_per_game DESC
LIMIT %s OFFSET %s
"""
```

Figure 19: Complex Query for Appearances

Below, you can see the query for appearances table and the result as the page exists below:

[Back to Player Features](#)

### Clubs with the Highest Average Goals Per Game

Only above average of 0.15 goals clubs are listed.

Club ID	Club Name	Average Goals Per Game
27	Bayern Munich	0.1921
338	Dynamo Kyiv	0.1680
18	Borussia Mönchengladbach	0.1634
124	Rangers FC	0.1621
234	Feyenoord Rotterdam	0.1608
131	FC Barcelona	0.1601
924	Istanbulspor	0.1600
383	PSV Eindhoven	0.1577
294	SL Benfica	0.1555
583	Paris Saint-Germain	0.1549

1 2 Next

Figure 20: Clubs with Highest Average Goals

This query shows average goals of clubs which are greater than 0.15 by joining appearances and clubs table and grouping the clubs.

## 4.5 Competitions Table

In this table, like the other ones, user may search for some features such as competition code or competition ID. Updating or deleting the existing data is also possible for each record using icons in the rightmost side. User can also insert new data with "Add Competitions" button.

Search by name, competition code, or competition ID...

[Sort and Filter Competitions](#) [Add Competition](#)

### Competitions

Competition ID	Competition Code	Name	Subtype	Type	Country ID	Country Name	Domestic League Code	Confederation	Update	Delete
BE1	jupiler-pro-league	jupiler-pro-league	first_tier	domestic_league	19	Belgium	BE1	europa		
BESC	belgian-supercup	belgian-supercup	domestic_super_cup	other	19	Belgium	BE1	europa		
CDR	copa-del-rey	copa-del-rey	domestic_cup	domestic_cup	157	Spain	ES1	europa		
CGB	efl-cup	efl-cup	league_cup	other	189	England	GB1	europa		
CIT	italy-cup	italy-cup	domestic_cup	domestic_cup	75	Italy	IT1	europa		
CL	uefa-champions-league	uefa-champions-league	uefa_champions_league	international_cup	-1	None	None	europa		
CLQ	uefa-champions-league-qualification	uefa-champions-league-qualification	uefa_champions_league_qualifying	international_cup	-1	None	None	europa		
DER	dfb-pokal	dfb-pokal	domestic_cup	domestic_cup	40	Germany	14	europa		

Figure 21: Competitions View Page

## 4.6 Club Games Table

In this table, we used buttons for insertion, deletion, sorting, filtering, and update operations and added search bar to reveal requested data such as all other tables. However, we added some extra features for this table.

Search by game ID, club\_id or club name...

Search

Sort and Filter Club Games

Club Game Top 10 Statistics

Add Club Game

Club Games

Game ID	Club ID	Club Name	Own Goals	Own Position	Own Manager Name	Opponent ID	Opponent Club Name	Opponent Goals	Opponent Position	Opponent Manager Name	Hosting	Is Win	Update	Delete
4055287	1083	FK Rostov	0	2	Valeriy Karpin	964	Zenit St. Petersburg	0	1	Sergey Semak	Home	0		
4055287	964	Zenit St. Petersburg	0	1	Sergey Semak	1083	FK Rostov	0	2	Valeriy Karpin	Away	0		
4055286	3725	Akhmat Grozny	1	7	Sergey Tashuev	1124	Fakel Voronezh	1	14	Dmitriy Pyatibratov	Away	0		
4055286	1124	Fakel Voronezh	1	14	Dmitriy Pyatibratov	3725	Akhmat Grozny	1	7	Sergey Tashuev	Home	0		
4055285	14589	FK Orenburg	1	8	Marcel Licka	932	Lokomotiv Moscow	4	10	Mikhail Galaktionov	Home	0		
4055285	932	Lokomotiv Moscow	4	10	Mikhail Galaktionov	14589	FK Orenburg	1	8	Marcel Licka	Away	1		
4055284	1186	Torpedo Moscow	1	16	Pep Clolet	232	Spartak Moscow	2	3	Guillermo Abascal	Home	0		
4055284	232	Spartak Moscow	2	3	Guillermo Abascal	1186	Torpedo Moscow	1	16	Pep Clolet	Away	0		

Figure 22: Clubs Games View Page

We added a navigation for some lists, which are "Top 10" Competitors With Maximum Goal Count, "Top 10 Clubs Successful in Away Games", and "Top 10 Managers With Winning Counts". To reveal this list user should click on the "Club Game Top 10 Statistics" button in the "Club Game View Page".

Back to Club Games

Club Game Top 10 Statistics

Top 10 Competitors With Maximum Goal Count

Top 10 Clubs Successful in Away Games

Top 10 Managers With Winning Counts

Back to Main Page

Figure 23: Top 10 Lists for Club Games

To obtain the first list, we have used "group by" for observing the each pair of clubs, and "having" for eliminate null club names since total goals for null names will cause a incorrect result. To reach club names, we have used "join" operation with games table. Then, we ordered in descending order and limited by 10 to create a "Top 10" list. Finally, we listed the results with home club names and away club names. While creating the second list, again we used join operation with games table to reach club names. Then, to evaluate their away performance, we grouped by away club name and took the count of winnings. After that, we ordered in descending order and limited by 10. Finally, we listed the results with club names.

Club Game Top 10 Statistics	
Top 10 Clubs Successful in Away Games	
Club Name	Winnings in Away Game
FC Barcelona	584
Real Madrid	564
Manchester City	560
Atlético de Madrid	554
Sevilla FC	548
Juventus FC	544
Bayern Munich	536
Manchester United	536
Liverpool FC	530
Chelsea FC	530

Figure 24: Top 10 Clubs Successful in Away Games

In the last list, we grouped by own manager names to evaluate each manager's performance separately. Since there is two records for each game, we have made operations with only own manager names. Then, we take the count of their winnings listing in descending order and limited the result with 10. Finally, most successful managers can be seen with their winning counts.

```
query = """SELECT g.home_club_name, g.away_club_name,
SUM(cg.own_goals + cg.opponent_goals) AS total_goals
FROM club_games cg
JOIN games g ON cg.game_id = g.game_id
GROUP BY g.home_club_name, g.away_club_name
HAVING g.home_club_name IS NOT NULL AND g.away_club_name IS NOT NULL
ORDER BY total_goals DESC
LIMIT 10"""
```

Figure 25: Complex Query for Club Games

In the query above, we are listing top 10 competitors which has maximum goal count. The list consists of home club name, away club name, and sum of the goal counts so far.

```
query = """SELECT cg.game_id, cg.club_id, c1.name, cg.own_goals, cg.own_position,
cg.own_manager_name, cg.opponent_id, c2.name, cg.opponent_goals,
cg.opponent_position, cg.opponent_manager_name, cg.hosting, cg.is_win
FROM club_games cg
LEFT JOIN clubs c1 ON cg.club_id = c1.club_id
LEFT JOIN clubs c2 ON cg.opponent_id = c2.club_id
WHERE cg.game_id = %s OR cg.club_id = %s OR
cg.opponent_id = %s OR c1.name LIKE %s OR c2.name LIKE %s
ORDER BY cg.game_id DESC LIMIT %s OFFSET %s"""
```

Figure 26: Complex Query for Club Games

In the query above, we join club games table with clubs table to use club name and opponent club game instead of club IDs. Then, we order by the game ID. The purpose of the query is revealing the result which consists of the user input partially or completely.

## 5 Challenges

During the project, there happened some difficulties such as deciding the database configuration method and deciding the create table queries. While creating tables, some foreign keys, primary keys and type decisions are tried and changed in case of an error coming from MySQL.

Additionally, because some tables had so many records, we encountered some slow loading for the pages and get 'lost connection to server' error in some cases. We handled slow loading with pagination but for the error to server, we had to get rid of some data. After reducing the millions of data to a proper size using random sampling from Pandas library of Python, we solved the problems.

Furthermore, some data was wrong like non existing club ids exists in players table, duplicate values etc. We cleaned them also using Pandas.