**Football Performance and Analytics System with MongoDB, MySQL, and Python : Group 16**


Alexandre Baptista
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64506@alunos.fc.ul.pt

Matei-Alexandru Lupașcu
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64471@alunos.fc.ul.pt

Lloyd DSilva
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64858@alunos.fc.ul.pt

Vram Davtyan
Informatics Department
Faculdade de Ciências da Universidade de Lisboa
Lisbon Portugal
fc64691@alunos.fc.ul.pt


## Project Description

This project (Github) is a **comprehensive football analytics system** designed to track and analyze various aspects of football matches, player performances, and team statistics. Using a combination of **MongoDB**, **MySQL**, and **Python**, the system enables the extraction of valuable insights from football data through advanced querying and aggregation techniques. These insights help teams, analysts, and enthusiasts make **data-driven decisions** related to player performances, match outcomes, and team rankings.

## Technical Implementation

- **MongoDB** and **MySQL** are utilized to store and manage the football data.
- **Python** is the primary tool for querying the databases, processing results, and visualizing data, offering an intuitive interface for interacting with the backend.

## Key Features and Data Components

1. **Player Statistics**:
    - Tracks individual player performances, including goals scored (e.g., head, foot), shot outcomes, and match details.
    - **Columns**: Player, Outcome, Distance, Body Part, match_id ...
2. **Match Details**:
    - Contains information about individual football matches, such as match date, league, attendance, and scores for both home and away teams.
    - **Columns**: Date, league, Round, Day, Attendance, home_id, away_id, score_away, score_home.
    - **Example Query**: "How many matches had attendance greater than 12,000?" This query filters matches where attendance exceeds 12,000.
3. **Club Information**:
    - Maintains statistics about football clubs, including their position in the league, matches played, wins, losses, goals scored, and points earned.
    - **Columns**: id, Pos, Matches, club_id, MP, year, name.
    - **Example Query**: "Rank teams based on their average attendance at home games." This query groups the data by `home_id`, calculates the average attendance, and sorts the results in descending order.
4. **Players Rankings**:
    - **Columns**: id, name, league ...
    - **Example Query**: "List shots where a goal was scored outside the 16m box, sorted by descending order." This aggregation pipeline counts goals made from outside the 16m box and ranks players based on total goals.

## Sample Queries

1. **Goals Scored by Head**:
    - Filters player shots that resulted in a goal using the head and counts the occurrences.
2. **Matches with Attendance Greater Than 12,000**:
    - Filters matches where the attendance exceeded 12,000 spectators.
3. **Goals Scored Outside the 16m Box**:
    - This query analyzes goals from outside the 16m box and ranks players by total goals scored.
4. **Team Ranking by Average Attendance**:
    - This query ranks teams based on the average attendance at their home games.

## Contributions

Throughout the development of this project, each student contributed to different sections, leveraging their individual strengths and skills. Key contributions include:

1. **Database Design**: Some team members focused on structuring the data in **MongoDB** and **MySQL** for efficient querying and aggregation, ensuring the database schema was optimized for performance.
2. **Query Development**: Other members collaborated on developing and optimizing complex **aggregation queries** to extract meaningful football statistics, such as goal attempts by body part, attendance analysis, and team rankings.
3. **Python Code:** Two team members are respnosible for writing the **Python scripts** that handle data processing, querying the databases, and generating insights. They also worked on creating **visualizations** to help interpret the data.

4. **Version Control and Collaboration**: Some students were focused on managing the project using **GitHub**, ensuring proper version control, collaboration, and smooth integration of different code sections from all team members.
5. **Performance Optimization**: All four members focused on **optimizing query performance**, ensuring that the system could handle large datasets efficiently, with fast response times for complex queries.

In addition to the individual contributions, the team held **1 physical meeting** to discuss project requirements, align tasks, and set milestones. Multiple **online calls** were also conducted to ensure continuous collaboration, address any challenges, and review progress.

## Known Errors

Data Type Mismatch and Large Number of Columns:
○ Problem: We experienced difficulty loading the dataset into MongoDB and SQL due to inconsistencies in data types and a large number of columns. Additionally, a large number of columns complicated the import process and impact database performance.
○ Causes:

- Different data formats and large number of columns in the original sources of your football data.

○ Solution:

- Data Transformation
- Schema Mapping
- Column Selection

## Execution Time Comparison

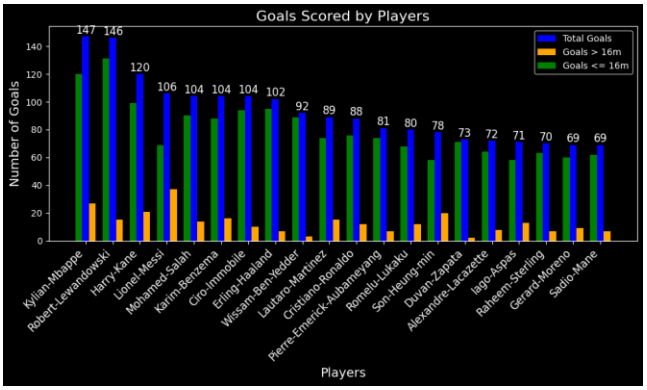
Q1.1
MySql : Execution time: 0.12903666496276855 seconds
Mongo : Execution time: 0.16304445266723633 seconds
Q1.2
MySql : Execution time: 0.0330202579498291 seconds
Mongo : Execution time: 0.037015438079833984 seconds
Q2.1
MySql : Execution time: 0.1870410442352295 seconds
Mongo : Execution time: 0.33606839179992676 seconds
Q2.2
MySql : Execution time: 0.02800583839416504 seconds
Mongo : Execution time: 0.04300713539123535 seconds

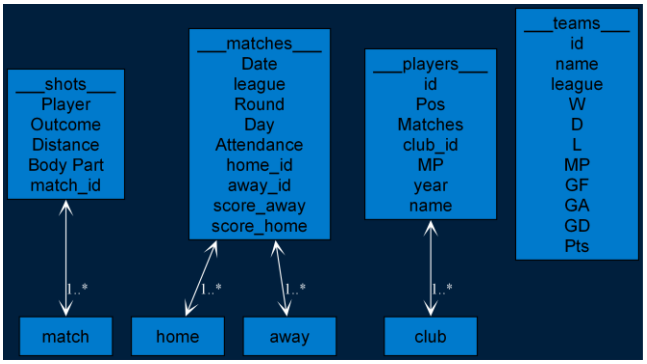
## Outcome

The project aims to provide **actionable insights** into football performance through **statistical analysis**. By leveraging **MongoDB** for advanced aggregation and **MySQL** for structured team data, the system delivers real-time insights into:

- **Player and team performance**
- **Match outcomes**
- **Fan engagement**

These insights allow for better decision-making regarding team strategies, player performance, and fan engagement, supporting both analysts and football enthusiasts.




## Potential Improvements for Phase 2

Better visaul representation for Diagram design, faster execution times, different queries used in football statistic, more efficent code structure.

**Links : Github**