First milestone of knowledge representation and reasoning project : Database

MVOMO ETO Wilfried
BAKHOLDIN Peter
DELPORTE Guillaume

July 9, 2023

1 Purpose of the Database

The goal of the MySQL database in the context of the provided schema is to store and manage data related to football (soccer). The database structure is designed to capture information about continents, cities, countries, teams, stadiums, players, managers, referees, and various statistics associated with players and teams.

By using this database, you can:

- Store information about continents, countries, cities, and their relationships.

- Maintain a list of teams, including their names, gender (male/female), and association with countries and stadiums.

- Manage player data, including their (respective identities) names, place of birth, date of birth, gender, country (nationality), and team affiliation. In our context, we assume that each player has a single nationality.

- Store information about managers and referees, including their (respective identities) names, place of birth, date of birth, gender, and country (nationality). In our context, we assume that each manager and referee has a single nationality.

- Track player statistics such as goals, assists, yellow cards, and red cards, associated with specific players and dates.

- Track team statistics such as the number of red cards, yellow cards, goals for, and goals against, associated with specific teams and dates.

2 Conceptual schema

continents:
Stores information about continents. Columns: id (primary key), name : name of the continent.
countries:
Stores information about countries. Columns: id (primary key), name : name of the country, continent_id (foreign key referencing continents.id).
cities:
Stores information about cities. Columns: id (primary key), name : name of the city, country_id (foreign key referencing countries.id).
teams:

Stores information about football teams. Columns: id (primary key), name : name of the team, is_male (specifies if the team is a male team), city_id (foreign key referencing cities.id).

stadiums:

Stores information about stadiums. Columns: id (primary key), name : name of the stadium, team_id (foreign key referencing teams.id), city_id (foreign key referencing cities.id).

players:

Stores information about football players. Columns: id (primary key), first_name : first name of the player, last_name :last name of the player, place_of_birth (foreign key referencing cities.id), birthday, country_id (foreign key referencing countries.id) : player nationality, team_id (foreign key referencing teams.id), is_male : player gender (specifies if the player is a male).

managers:

Stores information about football managers. Columns: id (primary key), first_name :first name of the manager, last_name :last name of the manager, place_of_birth (foreign key referencing cities.id), birthday, country_id (foreign key referencing countries.id) : manager nationality, team_id (foreign key referencing teams.id), is_male: manager gender (specifies if the manager is a male).

referees:

Stores information about referees. Columns: id (primary key), first_name : first name of the referee, last_name :last name of the referee, place_of_birth (foreign key referencing cities.id), birthday, country_id (foreign key referencing countries.id): referee nationality, is_male : referee gender (specifies if the referee is a male).

player_statistics:

Stores statistics of individual players. Columns: id (primary key), player_id (foreign key referencing players.id), statistics_date : match date, goals : the total number of goals scored by a player in a specific match, assists : the total number of assists realized by a player in a specific match, yellow_cards : the total number of yellow cards conceded by a player in a specific match, red_cards : the total number of red cards conceded by a player in a specific match, position: position that the player occupies on the pitch.

team_statistics:

Stores statistics of football teams. Columns: id (primary key), team_id (foreign key referencing teams.id), statistics_date : match date, number_of_red_cards : the total number of red cards conceded by a team in a specific match , number_of_yellow_cards : the total number of yellow cards conceded by a team in a specific match , goals_for: the total number of goals scored by a team in a specific match (it indicates the offensive performance of the team, reflecting the number of goals they have successfully scored), goals_against the total number of goals conceded by a team in a specific match(it indicates the defensive performance of the team, reflecting the number of goals opponents have scored against them).

Relationships:

- One-to-Many Relationship: Continent to Country

- One-to-Many Relationship: Country to City

- One-to-Many Relationship: City to Team

- One-to-Many Relationship: City to Player and Manager and Referee

- One-to-Many Relationship: Team to Stadium

- One-to-Many Relationship: Team to Player and Manager

- One-to-Many Relationship: Player to Player_Statistics

- One-to-Many Relationship: Team to Team_Statistics

3   Assumptions:

1. The football database is designed to store information about continents, countries, cities, teams, stadiums, players, managers, referees, player statistics, and team statistics.

2. Each continent can have multiple countries, and each country belongs to a single continent.

3. Each country can have multiple cities, and each city belongs to a single country.

4. Each city can have multiple teams, and each team belongs to a single city.

5. Each team can have multiple stadiums, and each stadium belongs to a single team and city.

6. Each player, manager, and referee has a unique ID and belongs to a country and city based on their place of birth.

7. Players and managers are associated with a team, while referees are not associated with any specific team.

8. Player statistics and team statistics are recorded with a specific date and associated with the respective player or team.

9. The database assumes a one-to-many relationship between players and player statistics, as well as between teams and team statistics.

4   Integrity Constraints and Business Rules:

1. The *continents* table enforces the foreign key constraint with the *countries* table, ensuring that a continent cannot be deleted if there are countries associated with it.

2. The *countries* table enforces the foreign key constraint with the *continents* table, ensuring that a country cannot be deleted if it belongs to a continent.

3. The *cities* table enforces the foreign key constraint with the *countries* table, ensuring that a city cannot be deleted if it belongs to a country.

4. The *teams* table enforces the foreign key constraint with the *cities* table, ensuring that a team cannot be deleted if it belongs to a city.

5. The *stadiums* table enforces the foreign key constraints with the *teams* and *cities* tables, ensuring that a stadium cannot be deleted if it belongs to a team or city.

6. The *players* table enforces the foreign key constraints with the *cities*, *countries*, and *teams* tables, ensuring that a player cannot be deleted if they have associated records in the *player statistics* table and that their place of birth, country, and team exist.

7. The *managers* table enforces the foreign key constraints with the *cities*, *countries*, and *teams* tables, ensuring that a manager cannot be deleted if they have associated records in the *team statistics* table and that their place of birth, country, and team exist.

8. The *referees* table enforces the foreign key constraints with the *cities* and *countries* tables, ensuring that a referee cannot be deleted if they have associated records and that their place of birth and country exist.

9. The *player_statistics* table enforces the foreign key constraint with the *players* table, ensuring that player statistics cannot be deleted if the associated player does not exist.

10. The *team_statistics* table enforces the foreign key constraint with the *teams* table, ensuring that team statistics cannot be deleted if the associated team does not exist.

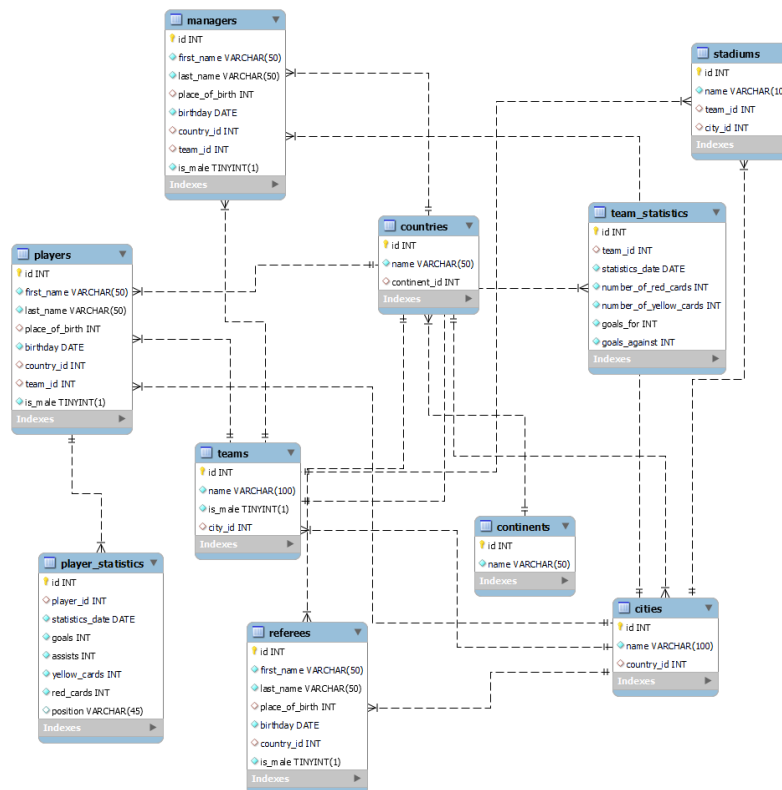# 5 Entity relationship diagram of our database:

Figure 1: Entity relationship diagram of our database

The contributors :

- BAKHOLDIN Peter, Purpose of the Database, conceptual schema, integrity constraints and business rules

- DELPORTE Guillaume, Purpose of the Database, integrity constraints and business rules, and conceptual schema and populating database

- Mvomo Eto Wilfried, conceptual schema, database schema and complex request dump file