**HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY**

**SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY**

**Ảnh có chứa văn bản, ký hiệu

Mô tả được tạo tự động**

DATABASE PROJECT REPORT

DATABASE LAB – IT3290E

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**FOOTBALL LEAGUE MANAGEMENT SYSTEM**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_­­­­­­­­­­­­­­­­­­­­­\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Team member Email*

Bùi Văn Thành 20200585 thanh.bv200585@sis.hust.edu.vn

Đỗ Tùng Lâm 20215218 lam.dt215218@sis.hust.edu.vn

*Class*

135409 – IT3290E

*Lecturer*

Vũ Tuyết Trinh

trinh.vutuyet@hust.edu.vn

*Wednesday, 08 March 2023*

Table of content

[I. Context 3](#_Toc684071752)

[II. Description 3](#_Toc1510549557)

[1. Data 3](#_Toc1444795224)

[2. Entities in database 3](#_Toc1091918070)

[a. Player 3](#_Toc1832634667)

[b. Club 3](#_Toc1960079288)

[c. Matches 3](#_Toc625852329)

[d. League year (Lyear) 4](#_Toc1269912215)

[The relation stores information related a season (league year) of a specific league 4](#_Toc1754698792)

[e. League 4](#_Toc2109494939)

[f. Stadium 4](#_Toc469831933)

[g. Match statistics 4](#_Toc220342339)

[h. is\_member 5](#_Toc1217947830)

[k. standing: 6](#_Toc1932617475)

[This relation does not follow the normalization, but it is so important that we cannot remove 6](#_Toc318592685)

[3. Diagram 6](#_Toc788291451)

[a. Entity-Relationship Diagram 6](#_Toc486299086)

[b. Relational Diagram 6](#_Toc1991061570)

[4. Functions 7](#_Toc708970610)

[a. Insertion 7](#_Toc152114774)

[b. Deletion 8](#_Toc521811354)

[c. Search 8](#_Toc889988321)

[d. Sorting 9](#_Toc1916290498)

[e. Trigger 9](#_Toc1805055170)

[f. Others function 9](#_Toc1546411386)

[5. Role of each user 10](#_Toc177338862)

[6. Index 12](#_Toc721486178)

[III. Work 13](#_Toc1825629429)

[IV. Conclusion 14](#_Toc1140478735)

1. **Context**

As far as you know, football is considered as a king sport in the world so that there are numerous people concern themselves with it. Therefore, we make a decision to build a Database managing information of league, footballers, clubs, matches through seasons in many Football Leagues whereby soccer fans are easy to find more useful information about a variety of things that is relevant to football.

1. **Description**

The system allows Admin, Team Leader, Customer and Guest to manage related information about, the league, clubs, players, matches in Football

Leagues. It includes four main functions: searching, sorting, deleting and inserting. There are four roles and each of them has the authority to use some of four functions, which is detailly described in section 2.4.

**Source code:** <https://github.com/thanhbv200585/Football-Management-Database>

## **1. Data**

All the data in the project was crawled in this website <https://fbref.com/en/> with BeautifulSoup. Beautiful Soup is a Python library for pulling data out of HTML and XML files. It works with your favorite parser to provide idiomatic ways of navigating, searching, and modifying the parse tree. After getting raw data, we use Pandas to clean the data, convert them into the expected form, then export them into csv files. Database is imported the data from with these file. Source code in folder data.

## **2. Entities in database**

Bold text is primary keys

Italic text is foreign keys

### **a. Player**

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | **player\_id** | Unique ID for a player |
| 2 | Player\_name | Player’s name |
| 3 | Nation | Player’s nationality |
| 4 | Position | Position of the player in line-up |
| 5 | Age | Player’s age |

### **b. Club**

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | **club\_id** | Unique ID for a club |
| 2 | club\_name | Club’s name |
| 3 | Nation | Name of the country |
| 4 | *Stadium\_id* | ID for club that the player is playing for |

### **c. Matches**

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | **Match\_id** | Unique ID for a match |
| 2 | *Home\_club\_id* | Home team’s id |
| 3 | *Away\_club\_id* | Away team’s id |
| 4 | *Lyear\_id* | ID for a year of the league |
| 5 | Score | Score of the match |
| 6 | Attendance | Number of Attendances in the stadium |
| 7 | Referee | The main referee’s name |

### **d. League year** (Lyear)

### The relation stores information related a season (league year) of a specific league

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | **lyear\_id** | Unique ID for a year of league |
| 2 | Season | Season year |
| 3 | Champion | The champion’s name |
| 4 | Number\_of\_squad | Number of teams in the season |
| 5 | Champion\_point | Number of points the champion got |
| 6 | Top\_scorer | The top scorer’s name |
| 7 | Goals | The number of goals top scorer scored |
| 8 | *League\_id* | ID for a league |

### **e. League**

|  |  |  |
| --- | --- | --- |
| No | Attribute | Note |
| 1 | **league\_id** | Unique ID for a league |
| 2 | League\_name | The league’s name |
| 3 | First\_season | The first season was held |
| 4 | Won\_most\_title\_club | The club’s name won the most titles |

### **f. Stadium**

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | **stadium\_id** | Unique ID for a league |
| 2 | Team | The team owns the stadium |
| 3 | Name | Stadium’s name |
| 4 | Capacity | Stadium’s capacity |

### **g. Match statistics**

|  |  |  |  |
| --- | --- | --- | --- |
| No | Attribute name | Note | Group |
| 1 | ***match\_id*** | Unique ID for a match | Primary key |
| 2 | ***Player\_id*** | Unique ID for a player |
| 3 | Short\_passes\_completed | Number of short passes completed (successful) | Passing |
| 4 | Short\_passes\_attempted | Number of short passes attempted |
| 5 | Long\_passes\_completed | Number of long passes completed (successful) |
|  | long\_passes\_attempted | Number of long passes attempted |
| 7 | Assists | Number of assists of a player in the match |
| 8 | Live | Number of live-ball passes | Pass types |
| 9 | Dead | Number of dead-ball passes |
| 10 | Free\_kick | Number of free kicks |
| 11 | TB | completed passes sent between back defenders into open space |
| 12 | Sw | Pass > 40 yard |
| 13 | TI | Number of throw-Ins taken |
| 14 | CK | Number of corner kicks taken |
| 23 | Def\_pen | Number of touches in penalty areas | Possession |
| 24 | Def\_3rd | Number of touches in defensive 1/3 |
| 25 | Mid\_3rd | Number of touches in middle 1/3 |
| 26 | Att\_3rd | Number of touches in attacking 1/3 |
| 27 | Succ | Dribbles completed successfully |
| 28 | Att | Dribbles attempted |
| 29 | Mis | Number of times a player when attempting to gain control of a ball |
| 30 | Dis | Number of times a player loss control of a ball after being tackled by an opposing player |
| 15 | tkl\_def\_3rd | Tackles in defensive 1/3 | Defensive actions |
| 16 | tkl\_mid\_3rd | Tackles in the middle 1/3 |
| 17 | tkl\_att\_3rd | Tackles in attacking 1/3 |
| 18 | dribblers\_tkl | Number of dribblers tackled |
| 19 | dribblers\_att | Total number of dribblers |
| 20 | intceptions |  |
| 21 | pass\_block | Number of passes blocked by a player |
| 22 | shot\_block | Number of shots blocked by a player |
| 31 | Total\_shot | The total number of shots | Shot |
| 32 | Goals | Number of goals |
| 33 | Shot\_on\_target | Number of shots on target |
| 34 | Penalty\_made | Number of penalties made (success) |
| 35 | Penalty\_attempted | Number of penalties attempted |
| 36 | Avg\_distance | The average distance of all shots |

### **h. is\_member**

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | ***club\_id*** | Unique ID for a club |
| 2 | ***Player\_id*** | Unique ID for a player |
| **3** | **Season** | ID for club that the player is playing for |
| 4 | Match\_played | Number of matches played for |
| 5 | Goals | Number of goals |
| 6 | Assists | Number of assists |

### **k. standing**:

### This relation does not follow the normalization, but it is so important that we cannot remove

|  |  |  |
| --- | --- | --- |
| No | Attribute name | Note |
| 1 | ***Lyear\_id*** | Unique ID for a league year (season) |
| 2 | ***Club\_id*** | Unique ID for a club |
| 3 | Match\_played | The number of matches this club played in this season |
| 4 | Points | Number of points |
| 5 | Pts | Goal difference |
| 6 | Goals | Total number of goals |
| 7 | Goals\_against | Total number of goals against |

## **3. Diagram**

### **a. Entity-Relationship Diagram**

Diagram, engineering drawing

Description automatically generated

Link diagram: [https://lucid.app/lucidchart/49455dcb-cd9f-4e71-a126-85101883cf5e/edit?page=0\_0&invitationId=inv\_826d5cde-5cfe-49e7-b596-1f6a2f3ccab5#](https://lucid.app/lucidchart/49455dcb-cd9f-4e71-a126-85101883cf5e/edit?page=0_0&invitationId=inv_826d5cde-5cfe-49e7-b596-1f6a2f3ccab5)

### **b. Relational Diagram**

Diagram, schematic

Description automatically generated

Link diagram: [https://lucid.app/lucidchart/849106cf-52f6-4004-a1f2-6d74f6b74101/edit?invitationId=inv\_226b5965-1ebd-45ea-ad37-a4ea69050964&page=0\_0#](https://lucid.app/lucidchart/849106cf-52f6-4004-a1f2-6d74f6b74101/edit?invitationId=inv_226b5965-1ebd-45ea-ad37-a4ea69050964&page=0_0)

## **4. Functions**

### **a. Insertion**

|  |  |  |
| --- | --- | --- |
| No | Function name | Description |
| 1 | insert\_stadium\_infor(team text,stadium\_name text,capacity integer) | Insert new stadium information |
| 2 | insert\_club\_infor(clb\_name text,country text) | Insert new club information |
| 3 | insert\_player\_infor(player\_name text,nation text,positions text,age integer) | Insert new player information |
| 4 | insert\_league\_infor(league\_name text,first\_season text,won\_most\_title\_club text) | Insert new league information |
| 5 | insert\_lyear\_infor(league\_name text,season text,num\_of\_squad integer,champion text,champion\_point integer,top\_scorer\_name text,goals integer) | Insert new league year information |
| 6 | insert\_match\_infor(leaugue\_name text,home text, away text, seasons text, score text, attendance integer, referee text) | Insert new match information |
| 7 | insert\_stats\_infor(...) // too many arguments | Insert new stat information for each player |

### **b. Deletion**

|  |  |  |
| --- | --- | --- |
| No | Function name | Description |
| 1 | delete\_stadium\_infor(stadium\_name text) | Delete stadium information by name |
| 2 | delete\_club\_infor(clb\_name text) | Delete club information by name |
| 3 | delete\_stats\_infor(playerName text) | Delete player stats by name |
| 4 | delete\_match\_infor(league\_name text,home text, away text, seasons text) | Delete match information |
| 5 | delete\_player\_infor(playerName text) | Delete player personal information (table player) |
| 6 | delete\_player\_record(playerName text) | Delete player record information (table is\_member) |
| 7 | delete\_lyear\_infor(league\_name text,seasons text) | Delete league year information by league name and season |
| 8 | delete\_league\_infor(league\_name text) | Delete league information |

1. **Search**

|  |  |  |
| --- | --- | --- |
| No | Function name | Description |
| 1 | search\_player\_infor(name text) | Search player information by name |
| 2 | search\_stadium\_infor(club\_name text) | Search stadium information by name |
| 3 | search\_player\_passing(name text) | Search player passing stats by name |
| 4 | search\_player\_passtype(name text) | Search player pass type stats by name |
| 5 | search\_player\_possession(name text) | Search player possession stats by name |
| 6 | search\_player\_defensive\_action(name text) | Search player defensive action stats by name |
| 7 | search\_player\_shoot(name text) | Search player shooting stats by name |
| 8 | search\_season\_infor(league\_name text,season text) | Search season by league name and season |
| 9 | search\_league\_history(league\_name text) | Search all seasons of a league |
| 10 | search\_first\_champion(name1 text) | Search the first champion in a league |
| 11 | search\_rank\_infor(league\_name text,name2 text) | Show standings of a league in a season |
| 12 | search\_player\_record(player\_name text, club\_name text) | Search player record of a player in a club |
| 13 | search\_club\_infor(clb\_name text) | Search club information |

### **d. Sorting**

|  |  |  |
| --- | --- | --- |
| No | Function name | Description |
| 1 | sort\_club(order\_by text,trend text) | Sort club information by column and trend |
| 2 | sort\_stadium(order\_by text,trend text) | Sort stadium information by column and trend |
| 3 | sort\_stats(order\_by text,trend text) | Sort stats information by column and trend |
| 4 | sort\_player\_record(order\_by text,trend text) | Sort player record by column and trend |
| 5 | sort\_league(order\_by text,trend text) | Sort league information by column and trend |
| 6 | sort\_player\_infor(order\_by text,trend text) | Sort player information by column and trend |
| 7 | sort\_season(order\_by text,trend text) | Sort season information by column and trend |

### **e. Trigger**

|  |  |  |
| --- | --- | --- |
| No | Function name | Description |
| 1 | update\_league() | Sort club information by column and trend |
| 2 | insert\_stats() | Sort stadium information by column and trend |
| 3 | delete\_stats() | Sort stats information by column and trend |
| 4 | update\_stats(); | Sort player record by column and trend |
| 5 | insert\_match() | Sort league information by column and trend |
| 6 | delete\_match(); | Sort player information by column and trend |

### **f. Others function**

Update\_standings\_from\_stratch(): update all standings from results each match in match table

## **5. Role of each user**

* **Admin:** grant all privileges on all tables, sequences, functions on database
* **Team Leader:** select, insert, update, delete on all tables, usage on the sequences, all privileges on all functions
* **Customer:** select all tables in schema
* **Guest:** select all tables except stats table

 Here is the list of users in our database.

|  |  |  |
| --- | --- | --- |
| **Username** | **Password** | **Role** |
| viewer1 | password1 | viewer |
| viewer2 | password2 | viewer |
| customer1 | password3 | customer |
| customer2 | password4 | customer |
| team\_leader1 | password5 | team\_leader |
| team\_leader2 | password6 | team\_leader |
| admin1 | admin1 | admin |

Graphical user interface, application, table

Description automatically generated

Demo code:

- Before updating:

Graphical user interface, application, Word

Description automatically generated

- Update:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

- After updating

Graphical user interface, application, Word

Description automatically generated

## **6. Index**

An index is used to retrive data from a database very fast, especially for finding records users need to search frequently.

In our project, we also apply and experiment some indexes in several tables to improve running time. Results are summarized in this table. View details in Index test.xlsx and Index.sql

Graphical user interface, application, Excel

Description automatically generated

1. **Work**

|  |  |  |
| --- | --- | --- |
| **Task** | **Đỗ Tùng Lâm** | **Bùi Văn Thành** |
| Find topic | x | x |
| Design entities in database | x | x |
| Design entity-relationship diagram | x | x |
| Design relational diagram | x | x |
| Find and crawl sample data |  | x |
| Create.sql (create table) script |  | x |
| View.sql script | x |  |
| Delete\_function.sql script | x |  |
| Insert\_function.sql script | x |  |
| Sort\_function.sql script | x | x |
| Search\_function.sql script | x |  |
| Role.sql script |  | x |
| Index.sql script |  | x |
| Trigger.sql script |  | x |
| Demo.sql script | x | x |
| Other\_function.sql script |  | x |

1. **Conclusion**

After completing the projects, we have some invaluable experiences:

* Teamwork plays an important role in designing a DB
* Analyse the requirements of database, design the ERD, then transform into relational database.
* Wrap the queries into a variety of functions, use triggers to ensure and update the database when updating data, create roles for different the kinds of users.
* Make connections to database using Java

However, there are some limits that we need to improve in the near future such as design graphical user interface. In addition, we can integrate more tools or machine learning algorithms to analyze the data.