

FINAL PROJECT
ARSENAL FOOTBALL MANAGEMENT SYSTEM

For

ITMD_510_04

Submitted by

Malak Desai (A20356870)

Classes:

1. Login
2. Team
3. Players
4. Tournaments (Parent Class)
5. EPL(Subclass)
6. UEFA(Subclass)
7. Player Records (Interface)

One Inheritance with 2 subclass:

Tournament is Parent Class and Premier League and Champions League is child class. The parent class will have attributes and methods that would be common to both the child classes like Tournament_ID, and Team_ID. The child class will have attributes and methods that would be SPECIFIC to that particular class. For example the premier league class will have player appearances and player goals in the premier league whereas the champions' league class will have the player appearances and player goals in the champions' league.

One Interface with one implementation:

The players will achieve records. All the records pertaining to the particular tournament will show up when a player selects that tournament name.

Tables:

1. Team
2. Tournament
3. Players
4. Premier League
5. Champions League

Extra Credit Points Implementation:

1. Applied CRUD for more than 1 entity: A manager can add, view, update and delete players. A player in this context (logically) can only implement 1 CRUD option that is to view details.
2. Applied the MVC pattern.
3. Polymorphism is used by writing methods for overriding a class method.
4. GUI components used: DATE PICKER, COMBO BOX, CHECK BOX and Menu Bar (can select any file from the system)
5. Used CSS styling in 3 views (images shown below in document)

Design:

The Arsenal Football Club has a **manager** and various **players** in the team. The project basically allows the manager to enter the application with the help of his login credentials and perform the basic **CRUD** functionality of Create, Read, Update and Delete.

The manager can:

- a. Add Player

- b. View Player
- c. Update Player
- d. Delete Player
- e. Add Tournament
- f. View Tournament
- g. Update Tournament
- h. Delete Tournament

Similarly, a player can login with his own personal credentials and perform actions that is applicable to him. Now a player cannot add another player or delete any other player. Neither can he update his own details. The reason being if a player updates his own details then he might change the number of goals scored by him in every match. So logically in this project the player can only View his details. So here only 1 CRUD functionality will be applicable to the player.

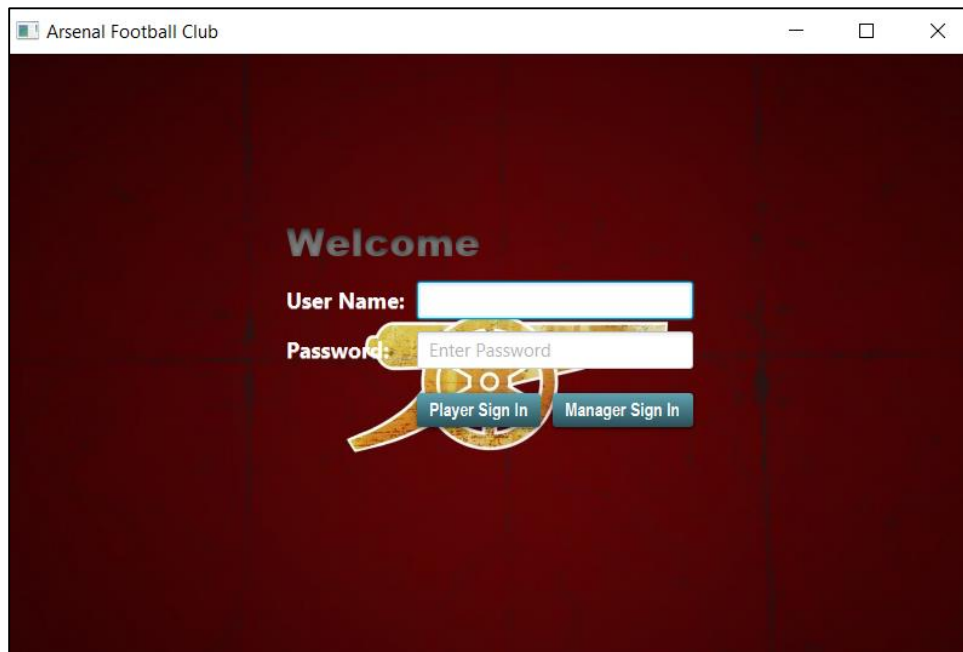
The player can:

- a. View Player Details
- b. View Tournament Details

Now let us start with the project.

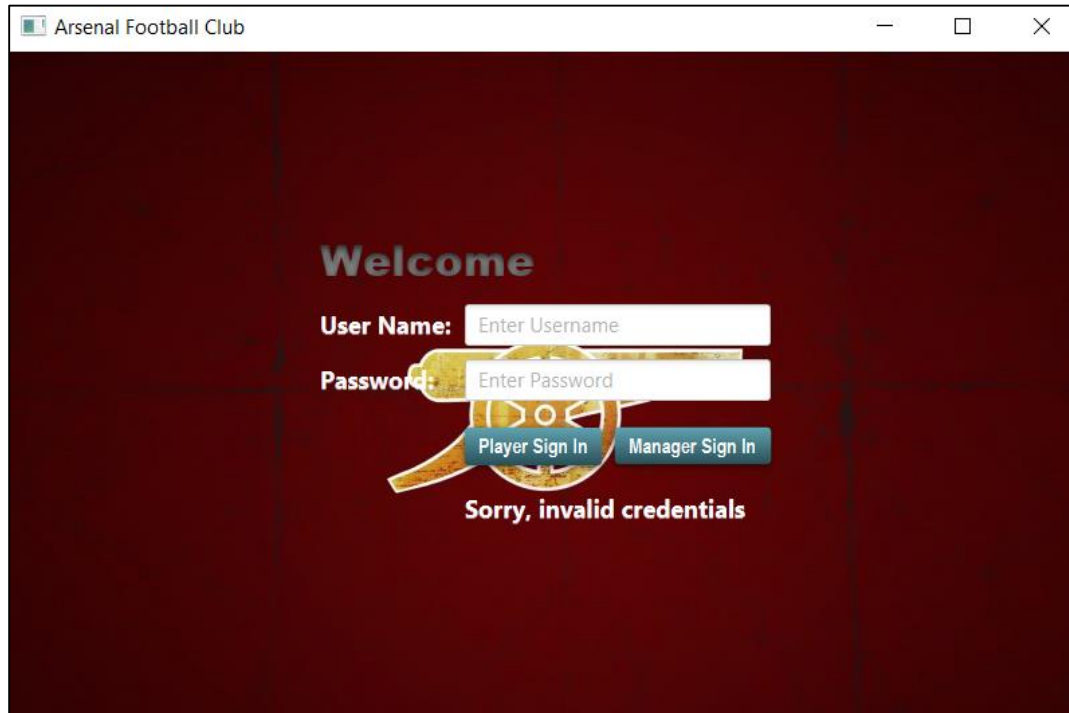
A. MANAGER

When the Login.java class file is run then a login page is displayed as shown below.

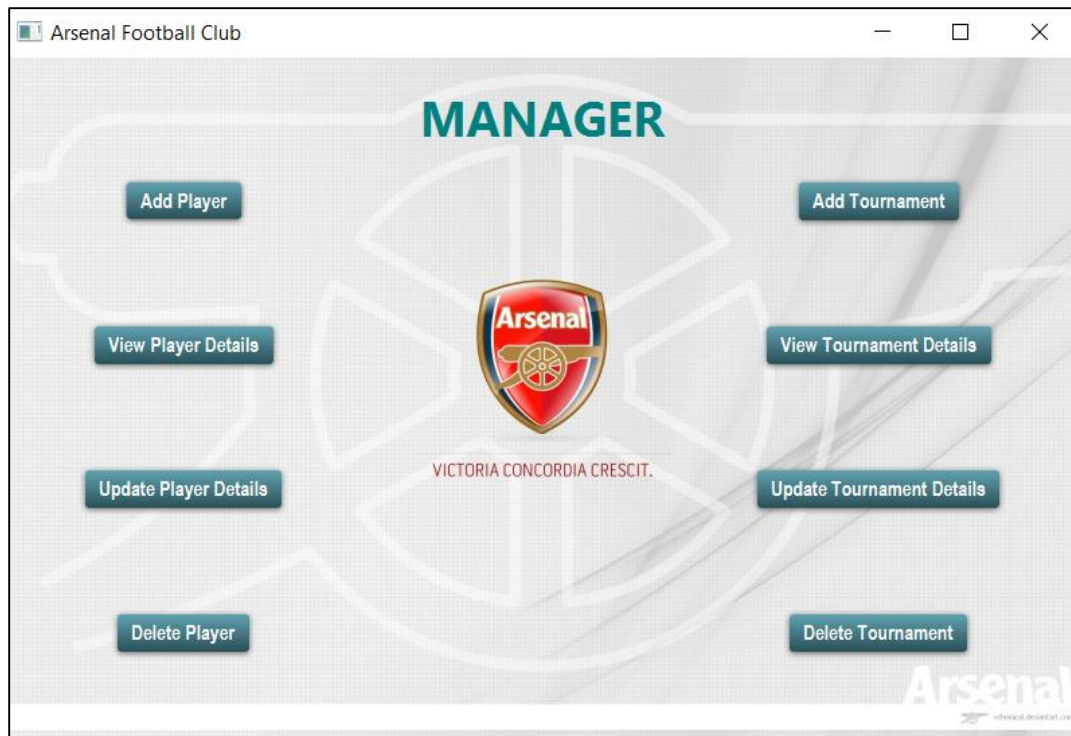


Now the correct login details for the manager is **USERNAME: Wenger** and **PASSWORD: 1001**. Another feature implemented here is that a manager can only sign in if he presses the Manager Sign In button. If he presses the Player Sign In button then he won't be able to login.

Let us say the manager put wrong details then the following screen will be displayed with the message saying "Sorry, invalid credentials."



When the manager puts in his correct details then he is guided to his manager page which is given below:



CRUD FUNCTIONALITY NUMBER 1: ADD PLAYER

Now when he clicks on Add Player button then the following screen pops up.

The screenshot shows a web application window titled "Arsenal Football Club". The main content area has a light gray background. At the top, there is a tab labeled "Player Information". Below the tab, the text "Player Table" is displayed. The form contains several input fields: "TEAM ID", "PLAYER ID", "PLAYER NAME", "PLAYER TEAM", "DATE OF BIRTH" (with a calendar icon), "PLAYER AGE", and "PLAYER SALARY". At the bottom of the form, there are two buttons: "SUBMIT" and "BACK".

The manager enters the player details that need to be added to the Players Table in the MySQL database server.

Arsenal Football Club

Player Information

Player Table

TEAM ID: 1

PLAYER ID: 111

PLAYER NAME: Ozil

PLAYER TEAM: Arsenal

DATE OF BIRTH: 10/15/1988

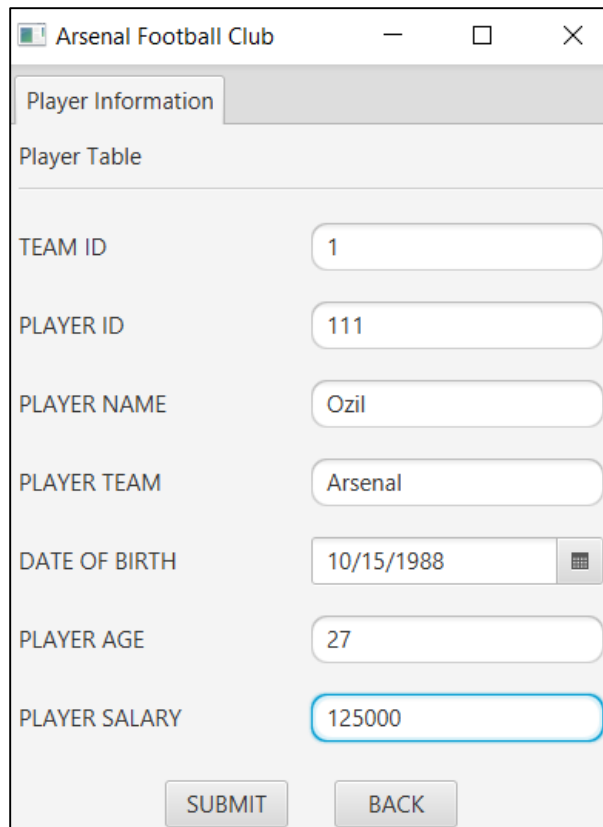
PLAYER AGE:

PLAYER SALARY:

SUBMIT

Sun	Mon	Tue	Wed	Thu	Fri	Sat
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

Here I have used the **DATEPICKER** function which lets me select the date from the calendar and **when I click on the text field of player age, the age is calculated and automatically displayed on click of a mouse.** After entering all the required details for the players and clicking on submit button then the data is stored in the database of MySQL under the players table.



A screenshot of a web application window titled "Arsenal Football Club". The window contains a form for entering player information. The form has a tab labeled "Player Information" and a section titled "Player Table". The form fields are as follows:

Field Label	Value
TEAM ID	1
PLAYER ID	111
PLAYER NAME	Ozil
PLAYER TEAM	Arsenal
DATE OF BIRTH	10/15/1988
PLAYER AGE	27
PLAYER SALARY	125000

At the bottom of the form, there are two buttons: "SUBMIT" and "BACK".

The data is saved in the table and the manager is brought back to his home page where he can carry out other functions.

The image for the new data that the manager added to the database is shown below.

Query 1 mdesai_uefa mdesai_epl mdesai_players x

Limit to 1000 rows

1 • `SELECT * FROM dbfp.mdesai_players;`

Result Grid Filter Rows: Edit: Export/Import:

	TEAM_ID	PLAYER_ID	PLAYER_NAME	PLAYER_TEAM	PLAYER_AGE	PLAYER_SALARY
▶	1	101	Cech	Arsenal	33	75000
	1	102	Debuchy	Arsenal	30	60000
	1	103	Gibbs	Arsenal	26	55000
	1	104	Mertesacker	Arsenal	31	75000
	1	105	Gabriel	Arsenal	25	50000
	1	106	Koscielny	Arsenal	30	80000
	1	107	Rosicky	Arsenal	35	95000
	1	108	Arteta	Arsenal	33	85000
	1	109	Welbeck	Arsenal	25	70000
	1	110	Wilshere	Arsenal	23	90000
	1	111	Ozil	Arsenal	27	125000
*	NULL	NULL	NULL	NULL	NULL	NULL

CRUD FUNCTIONALITY NUMBER 2: VIEW PLAYERS

When the manager clicks on View Player Details button on the manager pane then a page is displayed which shows the list of all the players in the team. The image as shown below.

Arsenal Football Club					
TEAM_ID	PLAYER_ID	PLAYER_NAME	PLAYER_TEAM	PLAYER_AGE	PLAYER_SALARY
1	101	Cech	Arsenal	33	75000
1	102	Debuchy	Arsenal	30	60000
1	103	Gibbs	Arsenal	26	55000
1	104	Mertesacker	Arsenal	31	75000
1	105	Gabriel	Arsenal	25	50000
1	106	Koscielny	Arsenal	30	80000
1	107	Rosicky	Arsenal	35	95000
1	108	Arteta	Arsenal	33	85000
1	109	Welbeck	Arsenal	25	70000
1	110	Wilshere	Arsenal	23	90000
1	111	Ozil	Arsenal	27	125000

CRUD FUNCTIONALITY NUMBER 3: UPDATE PLAYER DETAILS

Similar to add player, the manager can update any player details that he wants. On click of Update Player Details button the manager is guided to the same pane as add player details and there he can change any details he wants for example the player age and player salary. On clicking submit button the following data is changed.

The manager changed the data for player Ozil, making his age to 30 and his salary to 90000. The change is reflected in the below image.

Query 1 mdesai_uefa mdesai_epl mdesai_players x

Limit to 1000 rows

1 • SELECT * FROM dbfp.mdesai_players;

Result Grid Filter Rows: Edit: Export/Import:

	TEAM_ID	PLAYER_ID	PLAYER_NAME	PLAYER_TEAM	PLAYER_AGE	PLAYER_SALARY
	1	101	Cech	Arsenal	33	75000
	1	102	Debuchy	Arsenal	30	60000
	1	103	Gibbs	Arsenal	26	55000
	1	104	Mertesacker	Arsenal	31	75000
	1	105	Gabriel	Arsenal	25	50000
	1	106	Koscielny	Arsenal	30	80000
	1	107	Rosicky	Arsenal	35	95000
	1	108	Arteta	Arsenal	33	85000
	1	109	Welbeck	Arsenal	25	70000
	1	110	Wilshere	Arsenal	23	90000
	1	111	Ozil	Arsenal	30	90000
*	NULL	NULL	NULL	NULL	NULL	NULL

CRUD FUNCTIONALITY NUMBER 4: DELETE PLAYERS

The manager can remove/delete any player he wants. On clicking the Delete Player button on the home page, the following screen pops up where the details of the player to be deleted are entered.

Arsenal Football Club

Player Information | Tournament Information

Player Table

TEAM ID: 1

PLAYER ID: 111

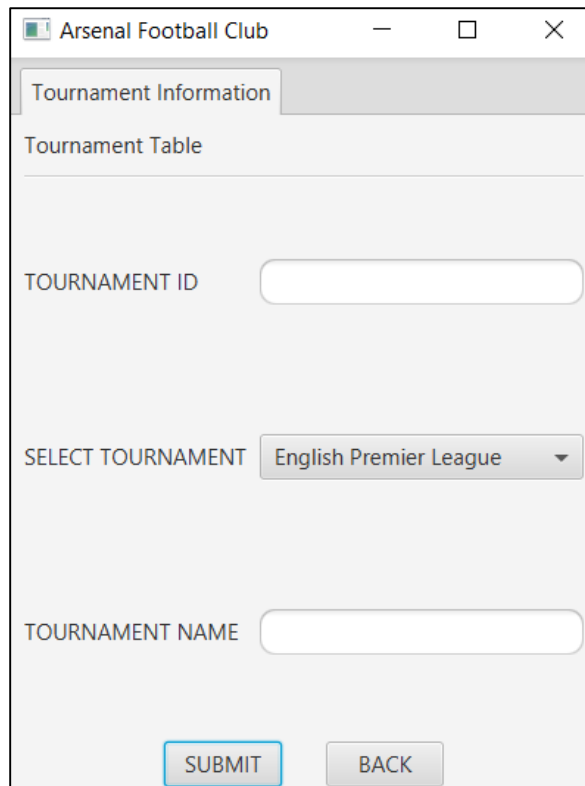
SUBMIT BACK

On clicking submit button the record of player with player ID 111 is deleted. The record is deleted from the database table. It is shown below.

Query 1 mdesai_uefa mdesai_epl mdesai_players x						
Limit to 1000 rows						
1 • SELECT * FROM dbfp.mdesai_players;						
Result Grid Filter Rows: Edit: Export/Import:						
	TEAM_ID	PLAYER_ID	PLAYER_NAME	PLAYER_TEAM	PLAYER_AGE	PLAYER_SALARY
	1	101	Cech	Arsenal	33	75000
	1	102	Debuchy	Arsenal	30	60000
	1	103	Gibbs	Arsenal	26	55000
	1	104	Mertesacker	Arsenal	31	75000
	1	105	Gabriel	Arsenal	25	50000
	1	106	Koscielny	Arsenal	30	80000
	1	107	Rosicky	Arsenal	35	95000
	1	108	Arteta	Arsenal	33	85000
	1	109	Welbeck	Arsenal	25	70000
	1	110	Wilshere	Arsenal	23	90000
*	NULL	NULL	NULL	NULL	NULL	NULL

Similarly, the manager can perform same actions on the Tournament. He can add tournament, view tournament details, update tournament dates and finally delete any tournament.

1. ADD TOURNAMENT



The screenshot shows a web application window titled "Arsenal Football Club". Inside the window, there is a tab labeled "Tournament Information". Below the tab, the text "Tournament Table" is displayed. The form contains three input fields: "TOURNAMENT ID" with a text input box, "SELECT TOURNAMENT" with a dropdown menu currently showing "English Premier League", and "TOURNAMENT NAME" with a text input box. At the bottom of the form, there are two buttons: "SUBMIT" and "BACK".

Arsenal Football Club

Tournament Information

Tournament Table

TOURNAMENT ID

SELECT TOURNAMENT

✓ English Premier League

UEFA Champions League

FA Cup

TOURNAMENT NAME

SUBMIT

BACK

2. VIEW TOURNAMENT DETAILS

Arsenal Football Club

TOURNAMENT_ID	PLAYER_ID	EPL_APPS	EPL_GOALS	EPL_YC	EPL_RC
10001	101	15	0	4	1
10001	102	12	1	3	0
10001	103	118	2	12	5
10001	104	135	5	2	1
10001	105	12	0	3	1
10001	106	160	13	22	6
10001	107	170	19	11	2
10001	108	107	14	8	4
10001	109	25	4	1	0
10001	110	100	6	15	2

Back

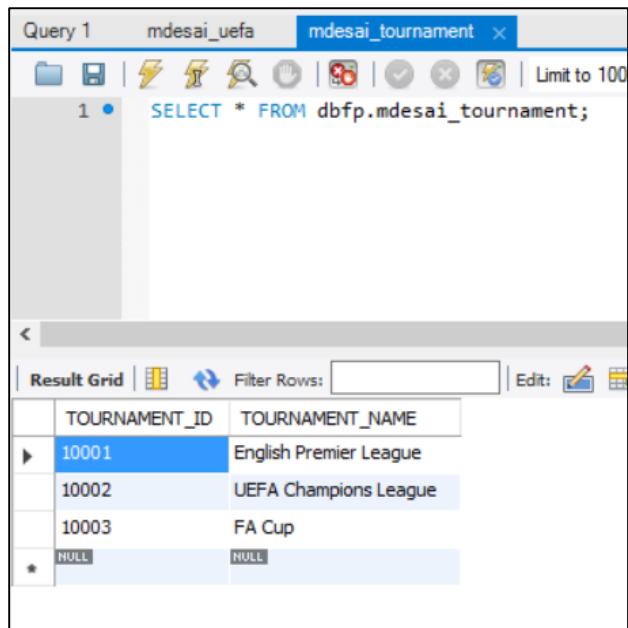
Next

3. UPDATE TOURNAMENT DETAILS

The tournament details that need to be updated like the date or the fixture is done by clicking on the Update Tournament Details button.

4. DELETE TOURNAMENT

Before deleting the tournament:



The screenshot shows a database query tool interface. At the top, there are tabs for 'Query 1', 'mdesai_uefa', and 'mdesai_tournament'. Below the tabs is a toolbar with various icons and a 'Limit to 100' option. The query editor shows the SQL statement: `SELECT * FROM dbfp.mdesai_tournament;`. Below the query editor is a 'Result Grid' section. It contains a table with two columns: 'TOURNAMENT_ID' and 'TOURNAMENT_NAME'. The table has four rows: 10001 (English Premier League), 10002 (UEFA Champions League), 10003 (FA Cup), and a row with NULL values. The first row is highlighted with a blue background.

TOURNAMENT_ID	TOURNAMENT_NAME
10001	English Premier League
10002	UEFA Champions League
10003	FA Cup
NULL	NULL

Details of tournament to be deleted are entered as shown below.

Arsenal Football Club

Tournament Information

Tournament Table

TOURNAMENT ID

TOURNAMENT NAME

SUBMIT BACK

After clicking on submit button the Tournament with ID 10003 is deleted from the table.

Query 1 mdesai_uefa mdesai_tournament

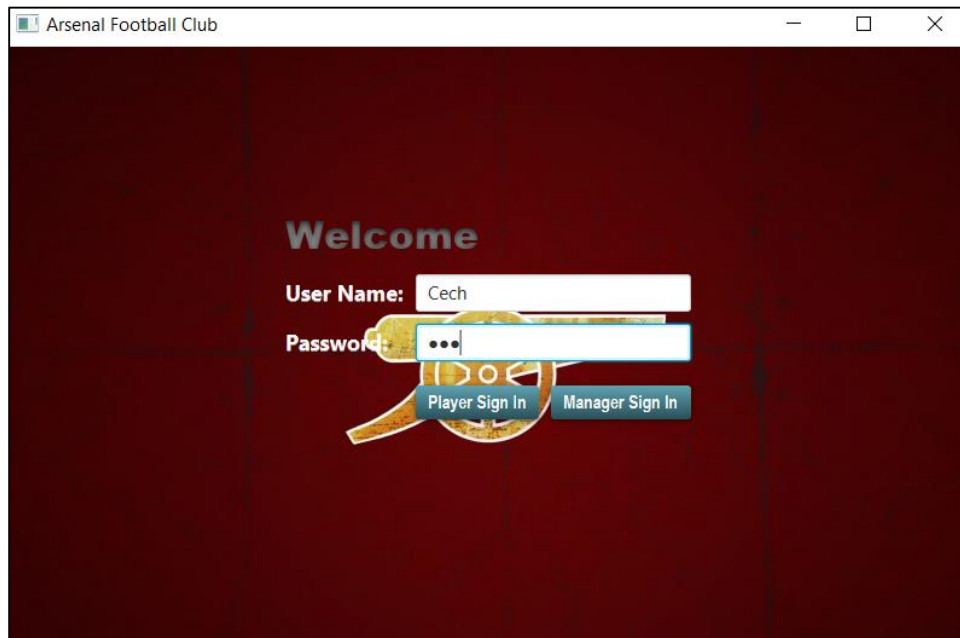
1 • `SELECT * FROM dbfp.mdesai_tournament;`

Result Grid

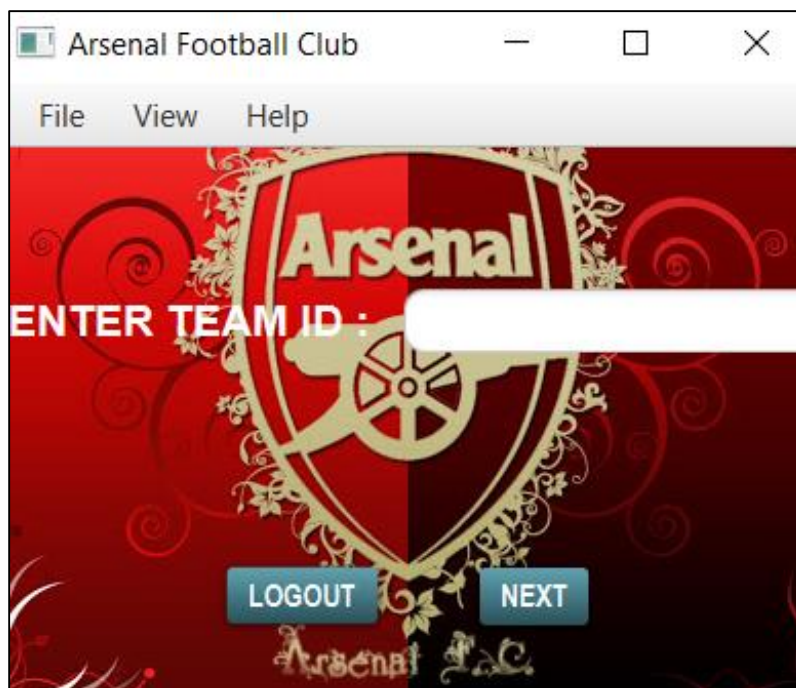
TOURNAMENT_ID	TOURNAMENT_NAME
10001	English Premier League
10002	UEFA Champions League
NULL	NULL

B. PLAYERS

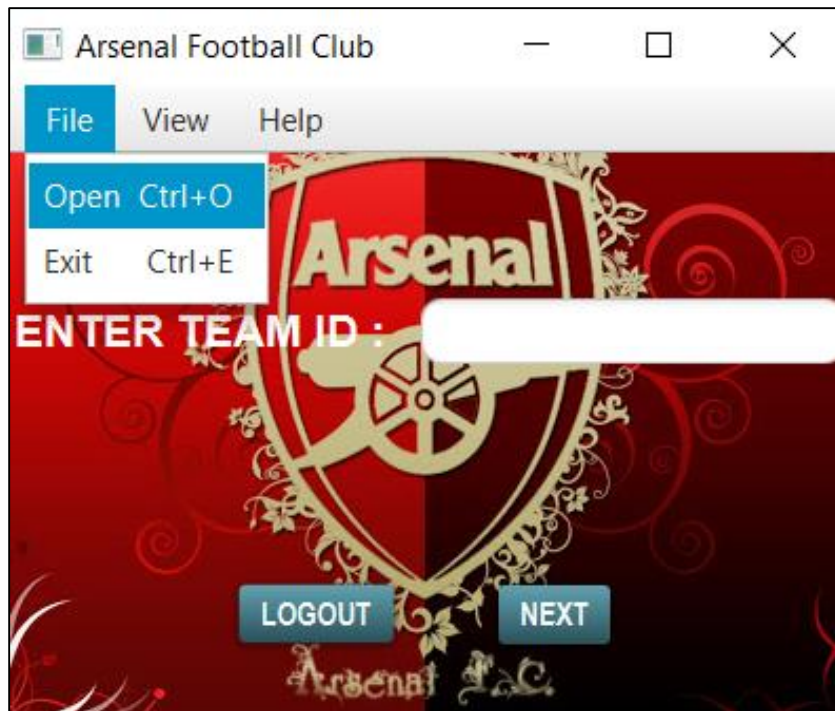
There are many players in the team so every player has its **unique** USERNAME and PASSWORD. The username for the players is their own name i.e. the Player Name and the password is their ID i.e. the Player ID. Let us consider that player with name **username CECH** and **password 101** is trying to login into the application.



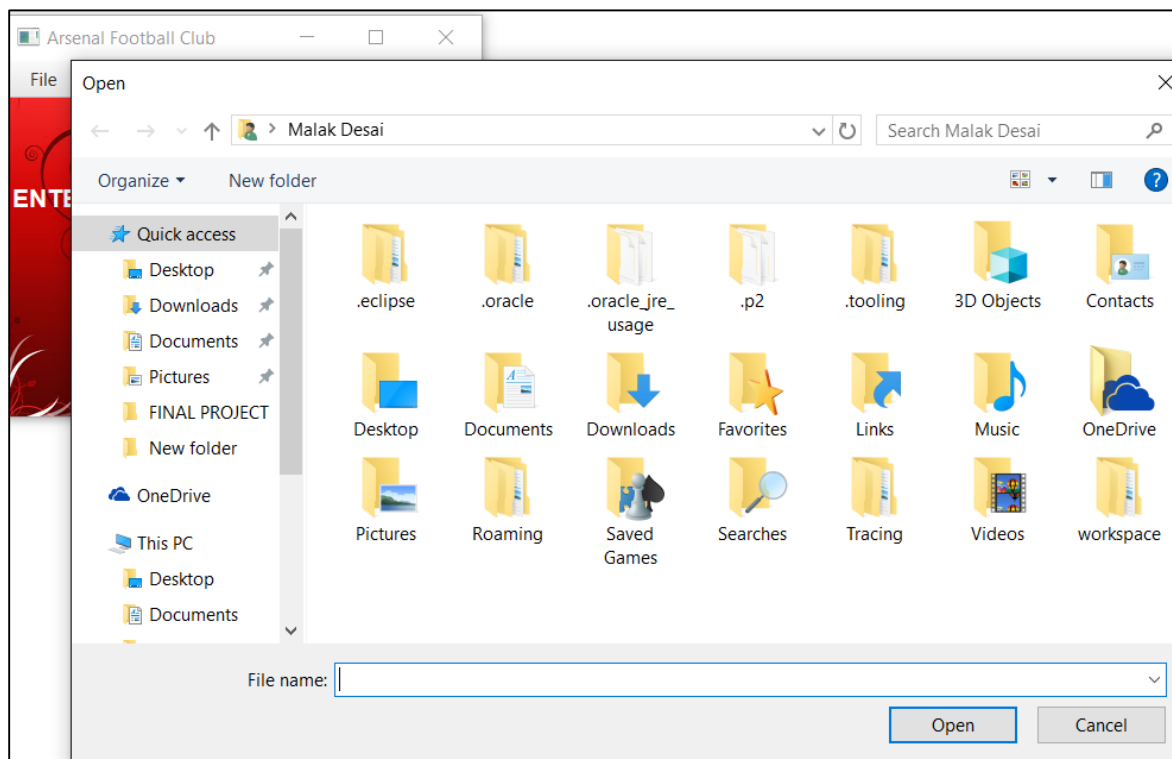
After he enters his correct details he will be prompted to enter the team ID for further verification whether that particular player is in that team or not.



Another feature of the player home page is that he can access any file from the system.



On clicking CTRL+O or clicking Open the following window pops up which shows the default system path of the computer and then the player can access any file he wants.



Experience working on the project:

Working on a JavaFX project for the first time was really challenging for me. I had to put in hard work learning the concepts of JavaFX and then implementing the same in my project. During the course of the project I learned many concepts about Java. It took me weeks to put in place my idea of phase 1 into effect. The end result though not completely fulfilled, I was able to implement more than I thought I would be able to. Overall experience was amazing and I would like to see more such exciting projects in the future.