Bilkent University

Computer Science Department

Object Oriented Software Engineering Project

STARS League

Design Report

Metehan Kaya, Babanazar Gutlygeldiyev, Yılmaz Berkay Beken, Kerem Ayöz

Course Instructor: Eray TÜZÜN

May 05, 2018

This report is submitted to the GitHub in partial fulfillment of the requirements of the Object Oriented Software Engineering Project, course CS319.

Contents

1. Introduction 1

2. Design Changes 1

3. Lessons Learned 3

4. User's Guide 3

4.1 System Requirements & Installation 3

4.2 How to Use 4

4.2.1 Controls 4

4.2.2 Game Objects 4

4.3 Screenshots and Menus 5

5. Incomplete Features 13

1. Introduction

From the beginning, we have been paying enough effort to finish all features we have promised. At the end, we were able to implement almost all of the features that we have included in the reports, except transfer feature we have added in the second iteration of Analysis Report. From now on, the users of the "STARS League" can create a specific account for each of them. Afterwards, he/she can choose a team and proceed to the tournament of the "STARS League". Throughout the tournament, the user will continue with group stage and knockout stage, unless gets eliminated in the group stage by not finishing in top two. After group stage, user has to beat all their opponents to be champion of the "STARS League". While playing the match, the user is able to change the team's tactics, and styles to increase the chance of winning the match. Moreover, the user will be able to change the players during the matches by making substitutions. If the user's team is able to finish both stages successfully and beat their opponents, this team will be awarded with "STARS League" cup.

2. Design Changes

We made some changes after the second design report.

1. Tournament
    a. Methods
        i. *goNextDay()*: Return type of this method is changed to pair of Match and integer since game has different logics for group stage and elimination stage. Here, a Match object which corresponds to the match that user is playing and an integer that corresponds to type of this match returns.
        ii. *checkChampionshipFailed()*: This method updates the value of *championshipFailed* attribute if the user's team has no chance to win the tournament after the last match.
    b. Attributes
        i. *championshipFailed*: A boolean that becomes true whenever the user's team has no chance to win the tournament.
        ii. *statusEliminationStage*: A boolean which shows if the game is in group stage or elimination stage. This attribute is defined for the simplicity, not to call methods again and again.

2. KnockoutTree
    a. Methods
        i. *playMatch(idMatch, isMyMatch)*: This method deals with two types of matches which are User vs Non-User and Non-User vs Non-User. Since *playMatch()* calls Match class's *matchSimulation()* for Non-User vs Non-User match, *isMyMatch* should be defined to separate these types of matches.
3. Match
    a. Methods
        i. *initAlgo()*: Initial values are assigned to attributes.
    b. Attributes
        i. *actionChance*: An integer that has a role in if an action is going to happen or not. If no action happens, then value of this variable increases which also increases the possibility of occurrence of next actions.
        ii. *timeHomeChance* & *timeAwayChance*: Integers that has a role in if an action is related to home team or away team. If an action is related to team home, then *timeHomeChance* decreases, and *timeAwayChance* increases which is to provide balance.
        iii. *actionGoalChance* & *actionInjuryChance* & *actionRedCardChance* & *actionYellowCardChance*: Each of them corresponds to chances of different types of action. Similar updates are applied on these attributes.
        iv. *tookRedCard[][]*: If a player gets a red card, then he cannot play a role in next actions. This attribute is used to prevent from inconsistency.
        v. *tookYellowCard[][]*: If a player gets 2 yellow card, then he gets a red card, and cannot play a role in next actions.
        vi. *playerPerformance[][]*: Since we defined injury in types of action, there should be a system which should not give high possibility of scoring a goal to injured players. If a player gets injured, then corresponding value decreases which also decreases the possibility of scoring a goal.
        vii. *playerNames[][]*: Since we allow the user to swap players during the match, we designed the Match class in a way that does not affected from this swapping in a negative way.

4. MatchPlayController
    a. Attributes
        i. currentMatchType: An integer that shows if the match that the user is playing a group match or elimination match.
        ii. currentMatchInfo: A pair of Match and integer that is compound of type of current match and type of it.

3. Lessons Learned

Throughout the semester, while implementing the project, we have learned how to utilize the features of the JavaFX Library by using Scene Builder. Moreover, we learned how to adjust system dependent features of application according to different Operating System. We experienced the benefits of predefined and fixed meeting hours that we arranged in the beginning. It served as backup meeting time for us in the busy weeks. Moreover, we learned the significance of the documentation and diagrams throughout the implementation. For instance, when we implement not so complex classes, we just typed the methods and the attributes already in the UML Diagrams. Last but not least, each of us has added yet another experience to his accumulation of experiences. It was more than a group integration experience for us.

4. User's Guide

In the User's Guide, we include a simple guide for installation and information about system requirements. Moreover, this section includes a guidance about how to use "STARS League".

4.1 System Requirements & Installation

"STARS League" is implemented in Java. Graphics will be implemented by using JavaFX libraries.
Minimum System Requirements

- Any platform that supports Java is supported
- 1.2 Ghz Dual Core processor
- 1 GB RAM or higher
- Minimum of 1400 x 900 screen resolution

Recommended System Requirements

- 2.0 Ghz Quad Core Processor
- 2GB RAM or higher
- Minimum of 1920x1080 resolution

Installation

"STARS League" is packaged into JAR file. So that our program does not require any installation procedure, just executing the "StarsLeague.jar" file is enough to play the game.

4.2 How to Use

In this section, we provide information about controls in the game and objects throughout the tournament in the game. The reason is that these are things the user will require in the "STARS League".

4.2.1 Controls

Mouse Left Click: Press the button.
Mouse Right Click: Drag the things.

4.2.2 Game Objects

Manager: That is the main object of the game and user manages the chosen team to win the "STARS League" tournament.
Player: Football player of the team.
Team: A team consists of players, manager and president.
Group Stage: Consists of 8 groups with each having 4 teams. 2 of them will pass the next stage.
Elimination: Consists of the elimination tree and teams that play with each other in elimination stage.

4.3 Screenshots and Menus

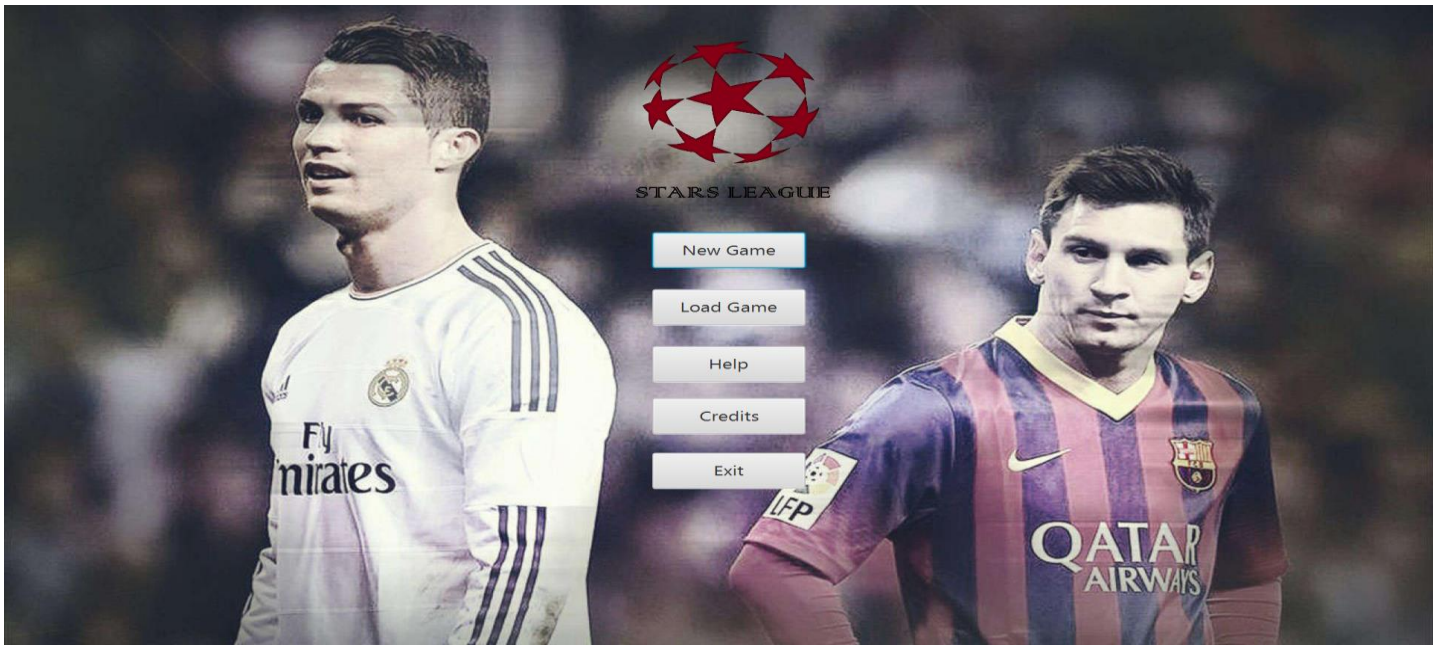

Figure 4.4.1 - Home Screen

This is the opening screen of the “STARS League”. We picked background picture with Cristiano Ronaldo and Lionel Messi, the two players that influenced many people to watch football matches. User can begin a new tournament, by clicking “New Game” button or load the previous game from that screen. Additionally, tips are displayed if the “Help” button is pressed. Player can exit from the game by pressing the exit button.

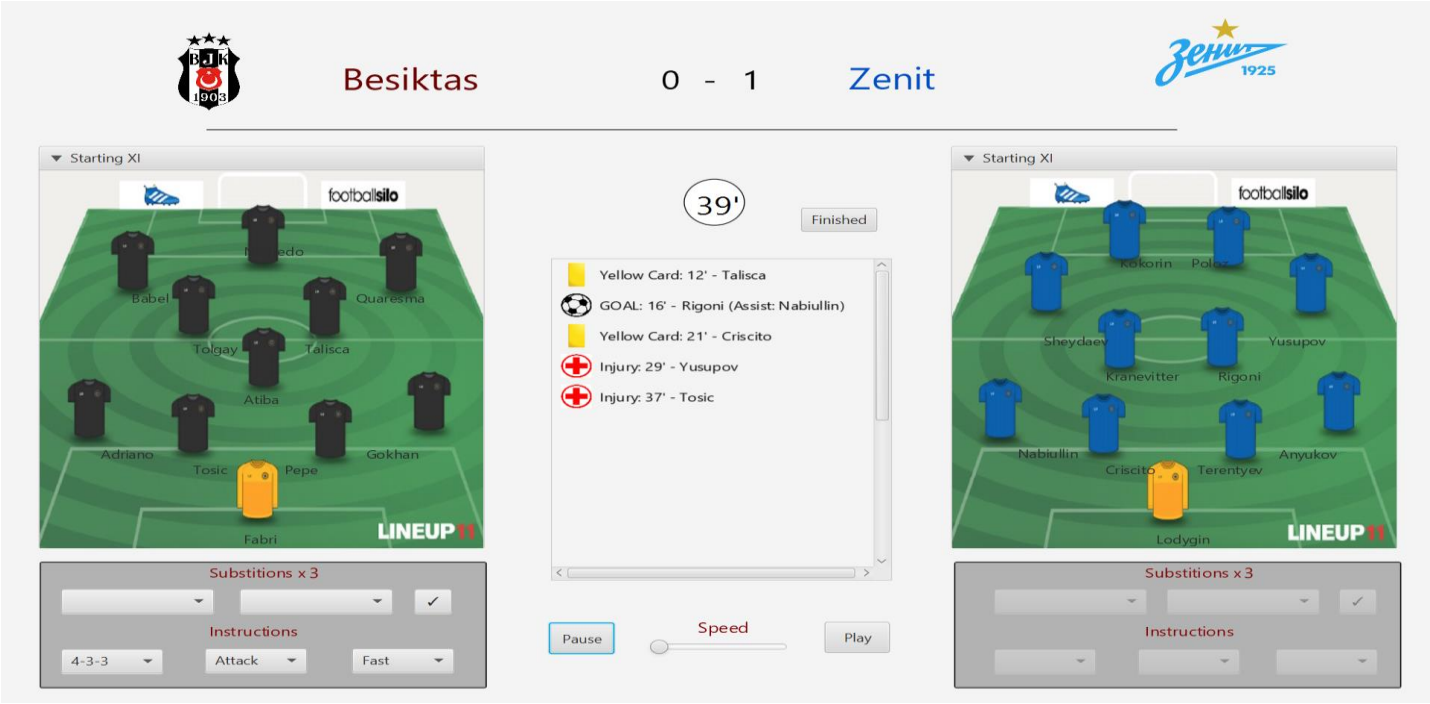

Figure 4.4.2 - Match Playing View

When the user starts a match, the tactics of his/her team are appeared on the below of the team's lineup. Actions which happened in the match (goals, assists, yellow card, red card) information are displayed with its minutes happened. The current line-ups appear in the below of the actions.

Figure 4.4.3 - Calendar View

When the user wants to see schedule with the approaching match(es), this screen will show up with team logos. The matches could be seen from that scene.

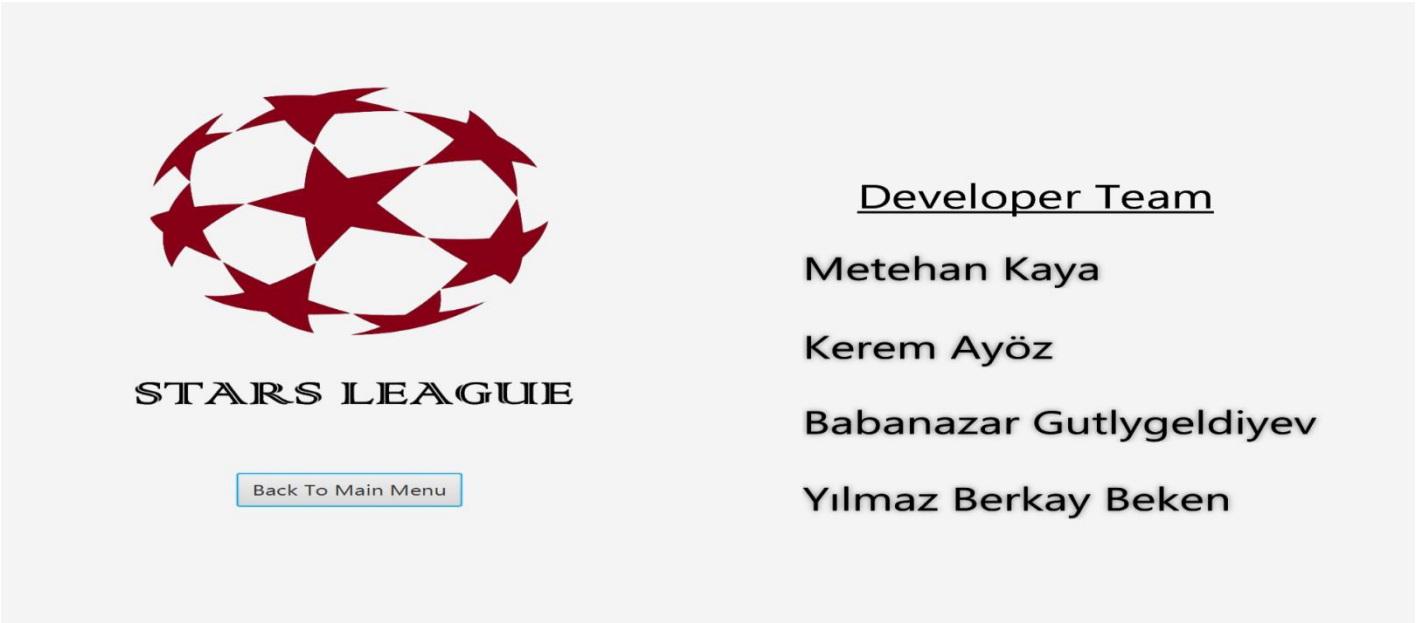

Figure 4.4.4 – Credits View

In this view, the user is welcomed with the credits given to the developers of “STARS League”. He/she is redirected to here from the Home Screen.

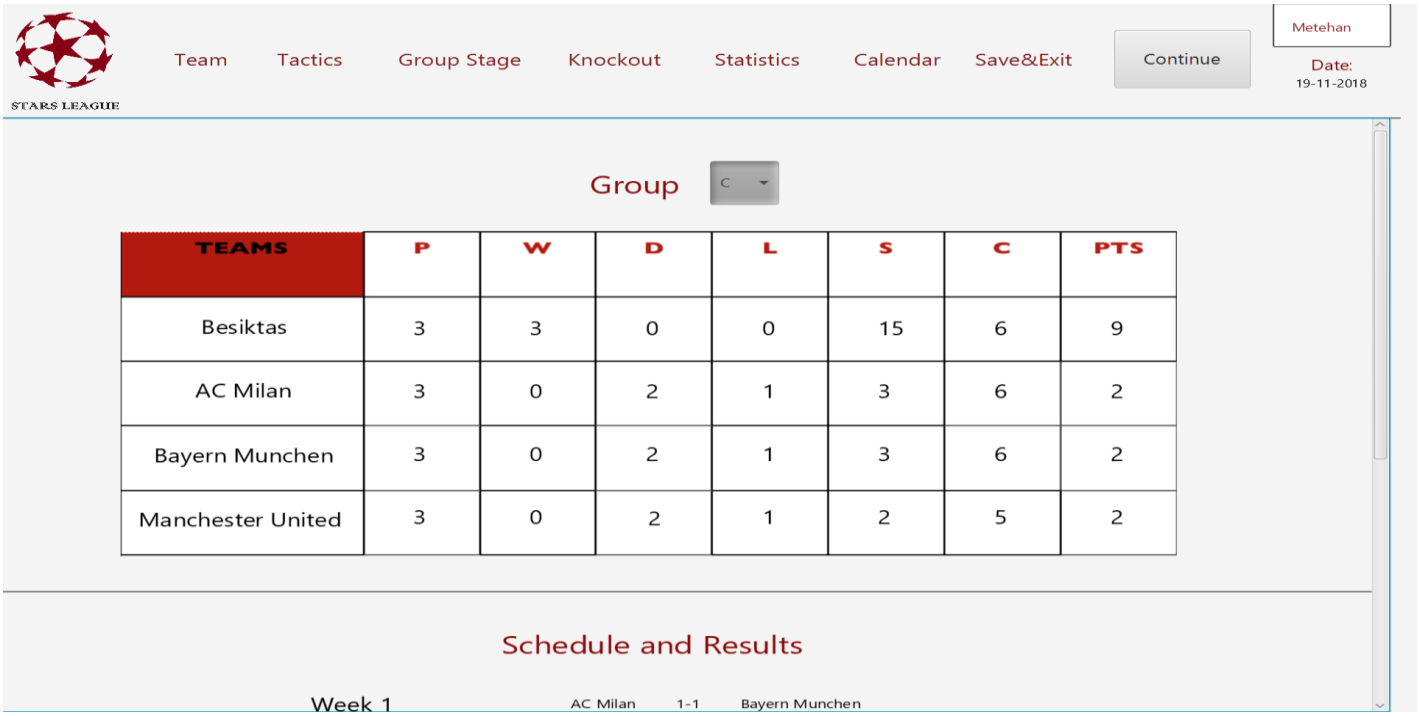

Figure 4.4.5.1 – Group View

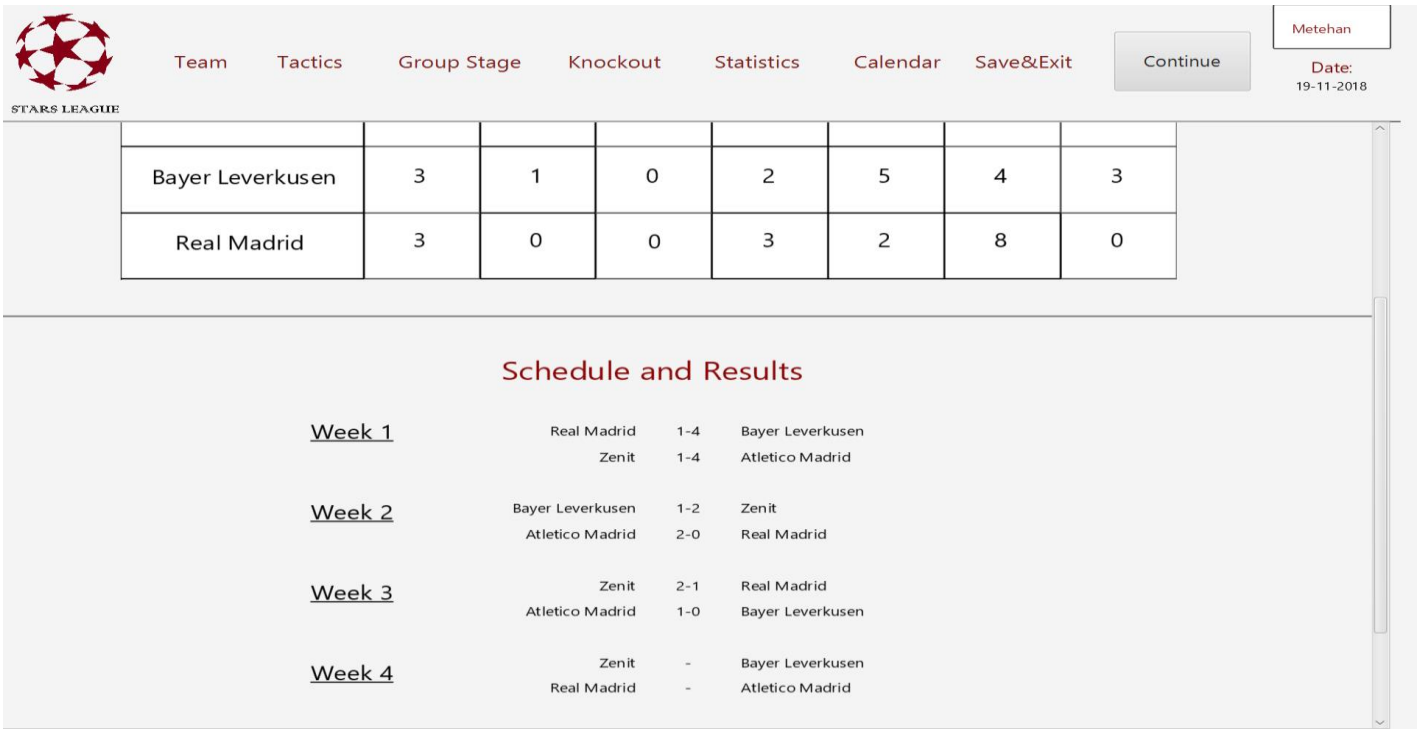

Figure 4.4.5.2 – Group View

Figures 4.4.5.1 and 4.4.5.2 include the views that show information about the group. Following the table of points writes information the matches of the user's team has played up to that time.

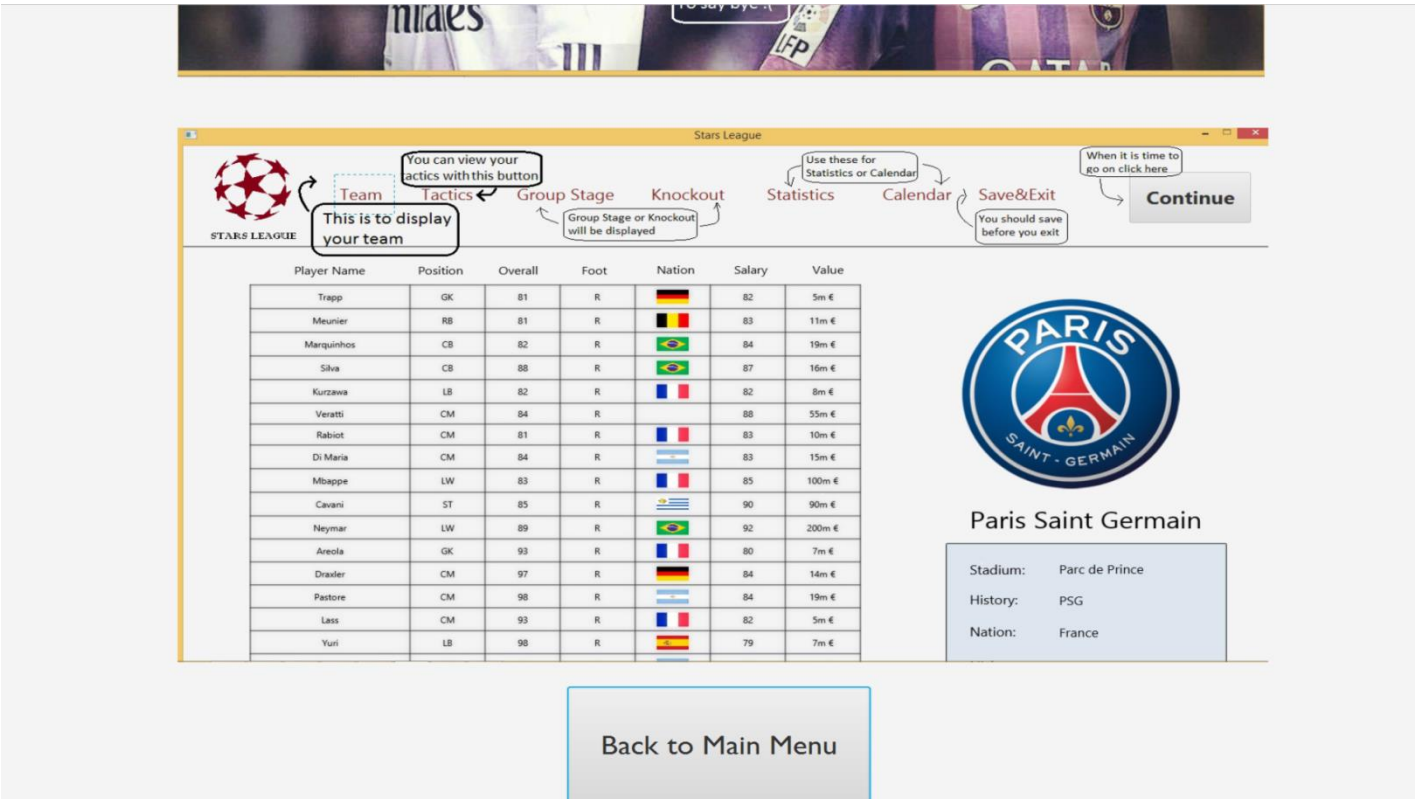

Figure 4.4.6 – Help View

In the figure above, Help View is shown, following a click on "Help" button in the Home Screen. Information about functions of the buttons is included in this view.

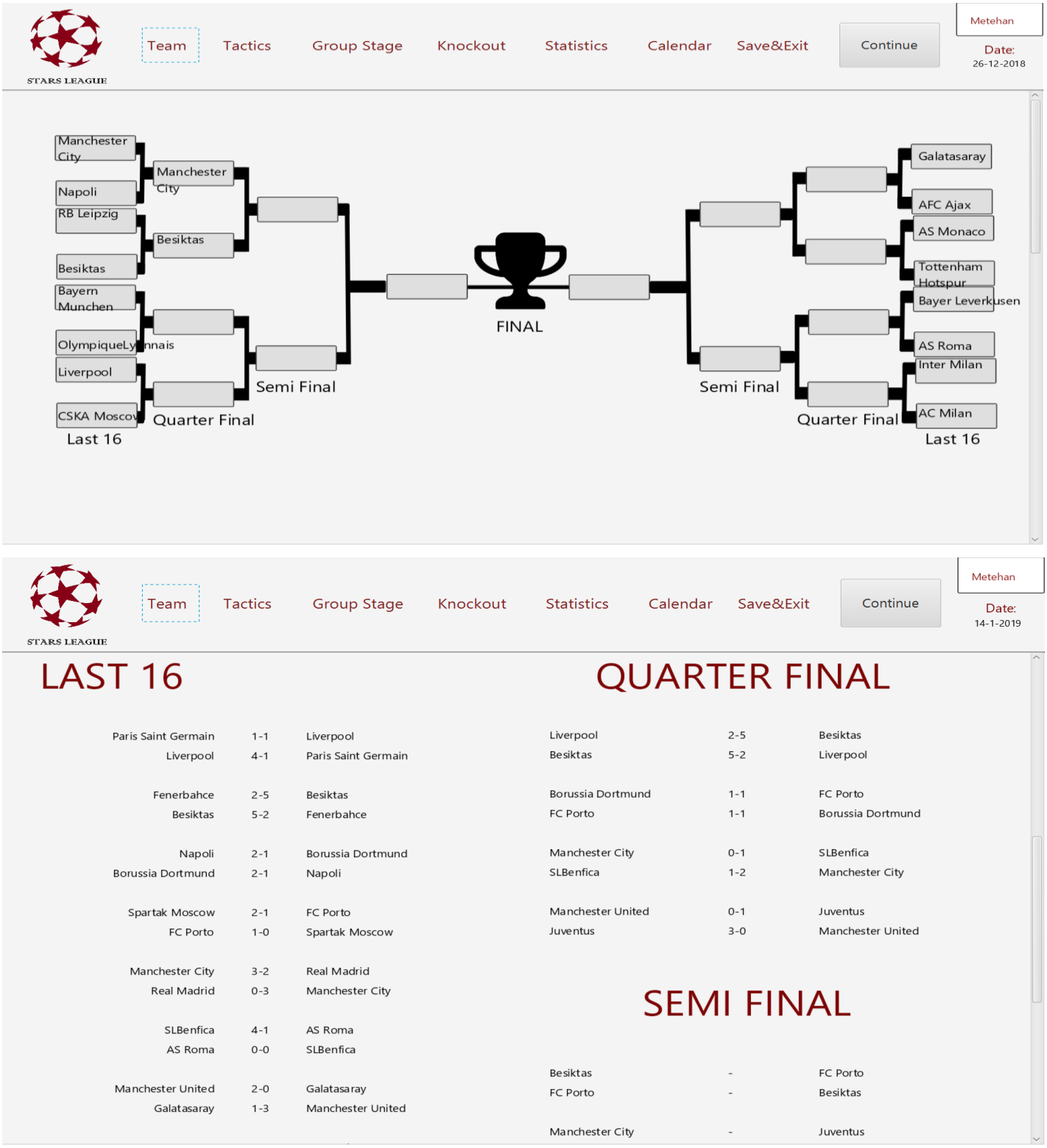

Figure 4.4.7 – Knockout View

This is the knockout view the user sees after his/her team is done with group stage. After each match, this view is updated according to the results of the match. In order to get the cup the user needs to be successful in this stage.

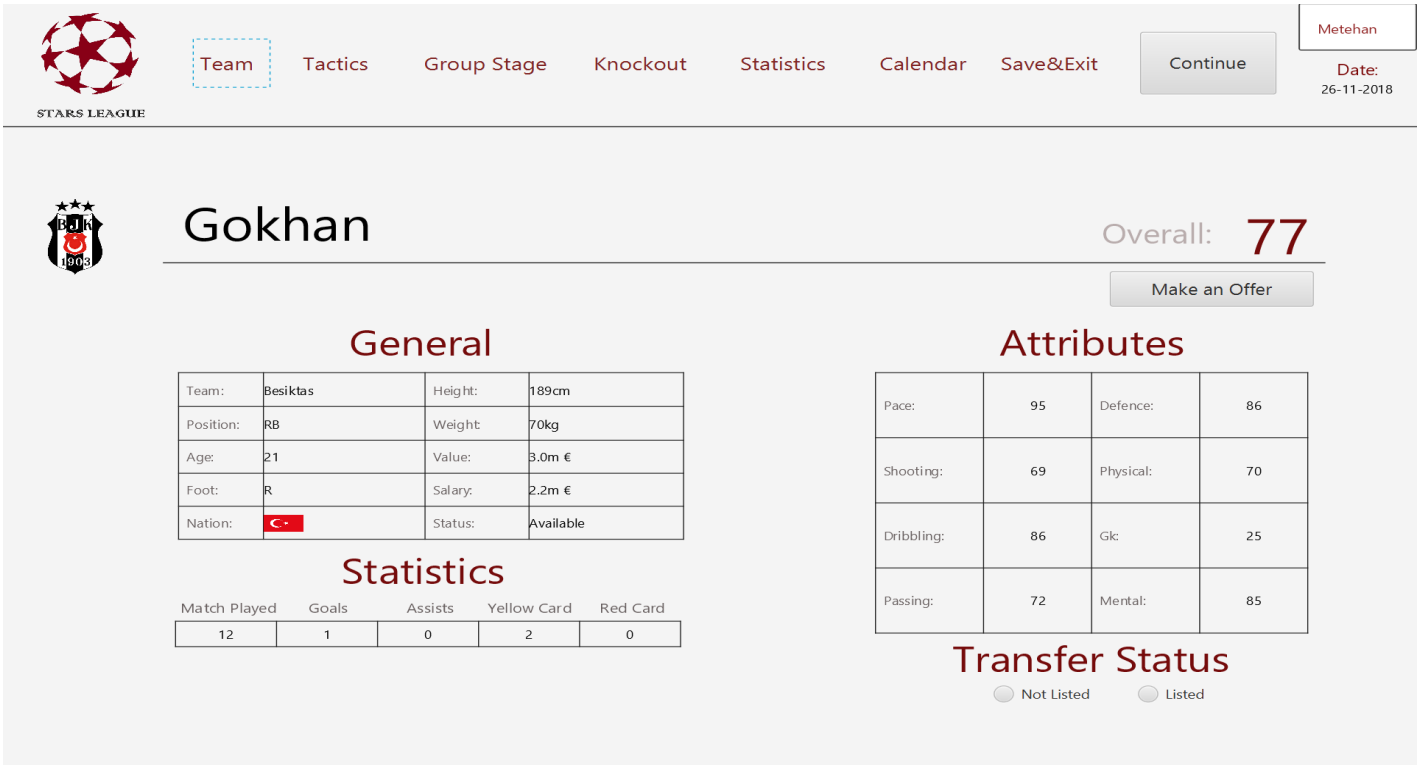

Figure 4.4.8 – Player View

This is the place where the user gets information of a player. He/she can see this view by clicking on the players' name. These statistics are updated after each match accordingly.

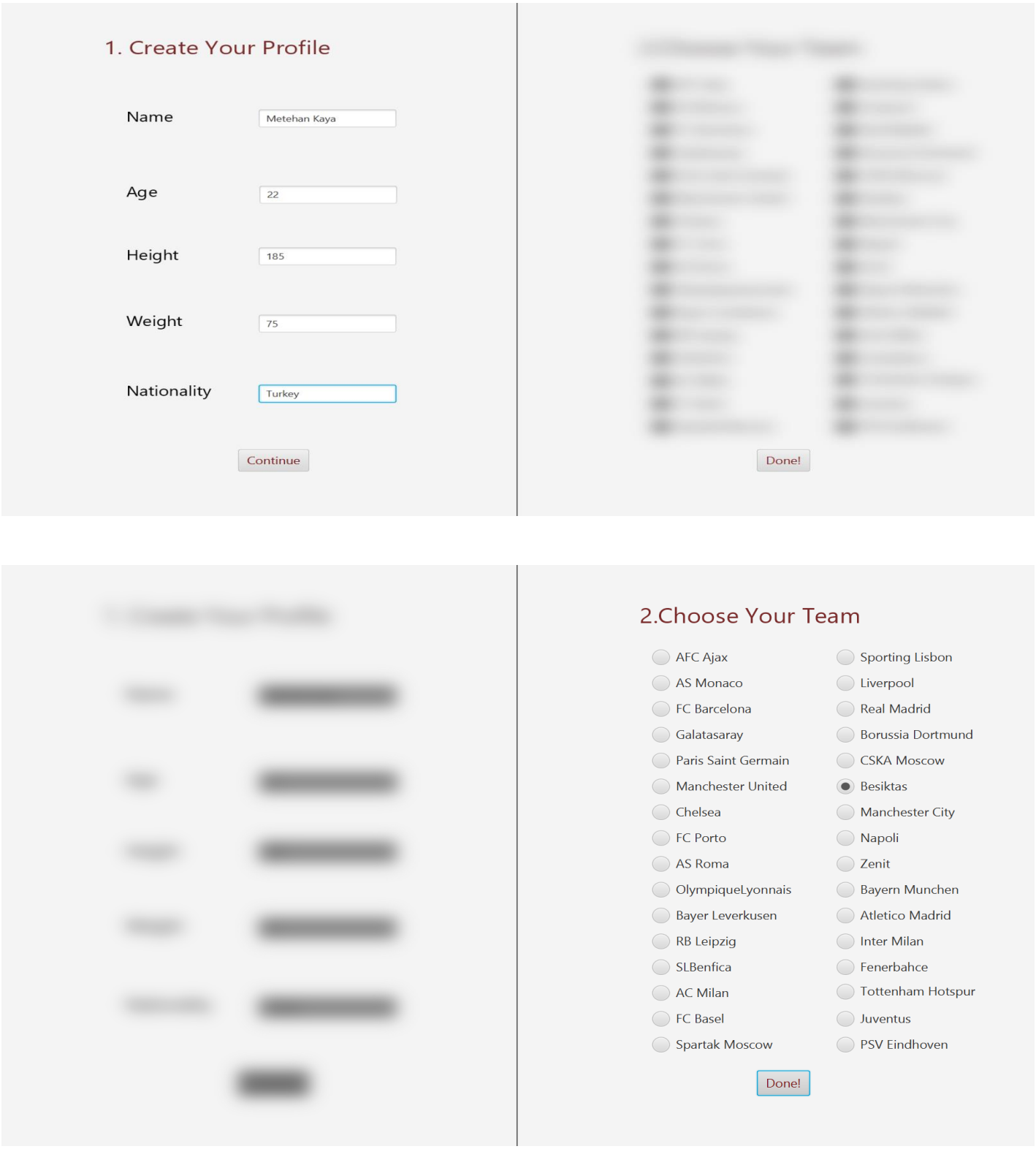

Figure 4.4.9 – Registration View

In this combined view of the Registration View, the user is asked to enter his/her information and pick a team among the options to win the tournament. Afterwards, the user will be directed to the team view.

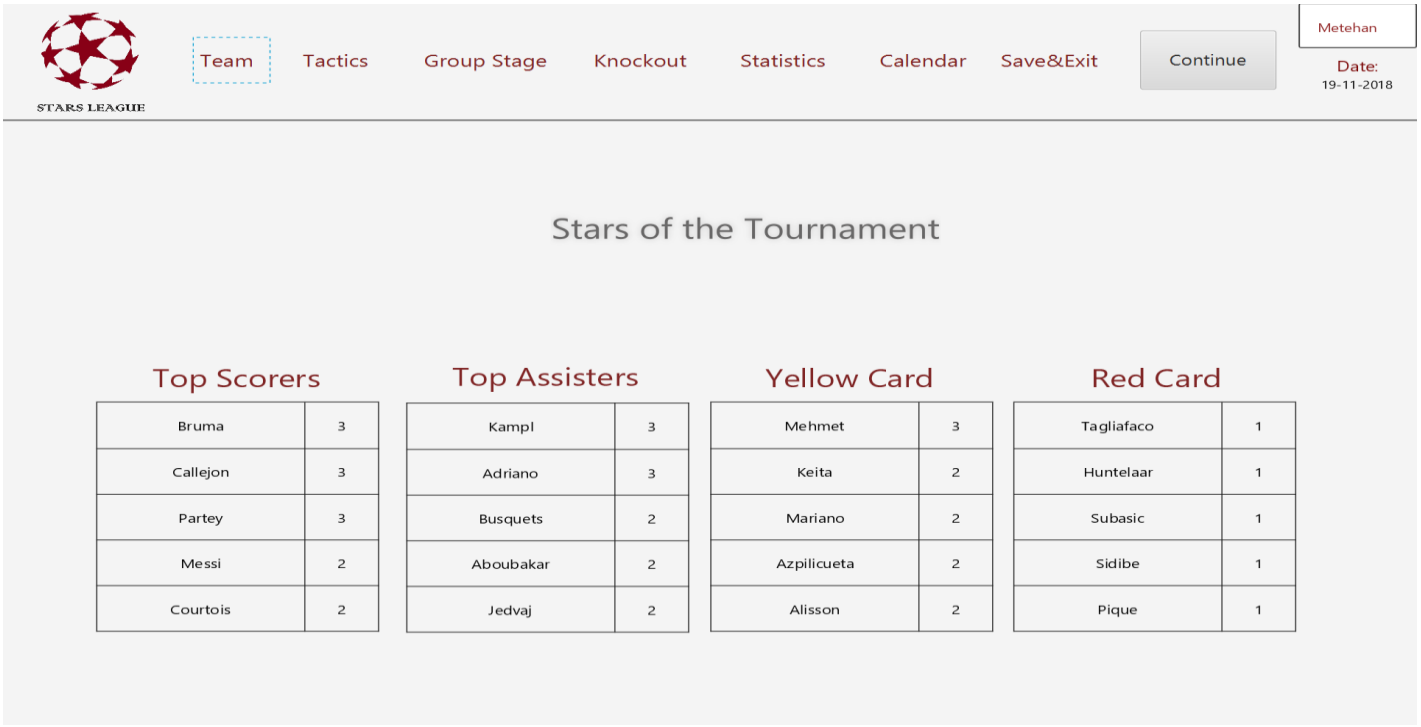

Figure 4.4.10 – Statistics View

In this view the user sees the information about the tournament's tops. These statistics are updated after each match accordingly.

Figure 4.4.11 – Tactics View

Figure 4.4.11 includes all information about the user's team's performance in a match. Moreover, the user can optimize his team by changing tactics, style and tempo of it in this view. Additionally, team's lineup can be arranged in this view.

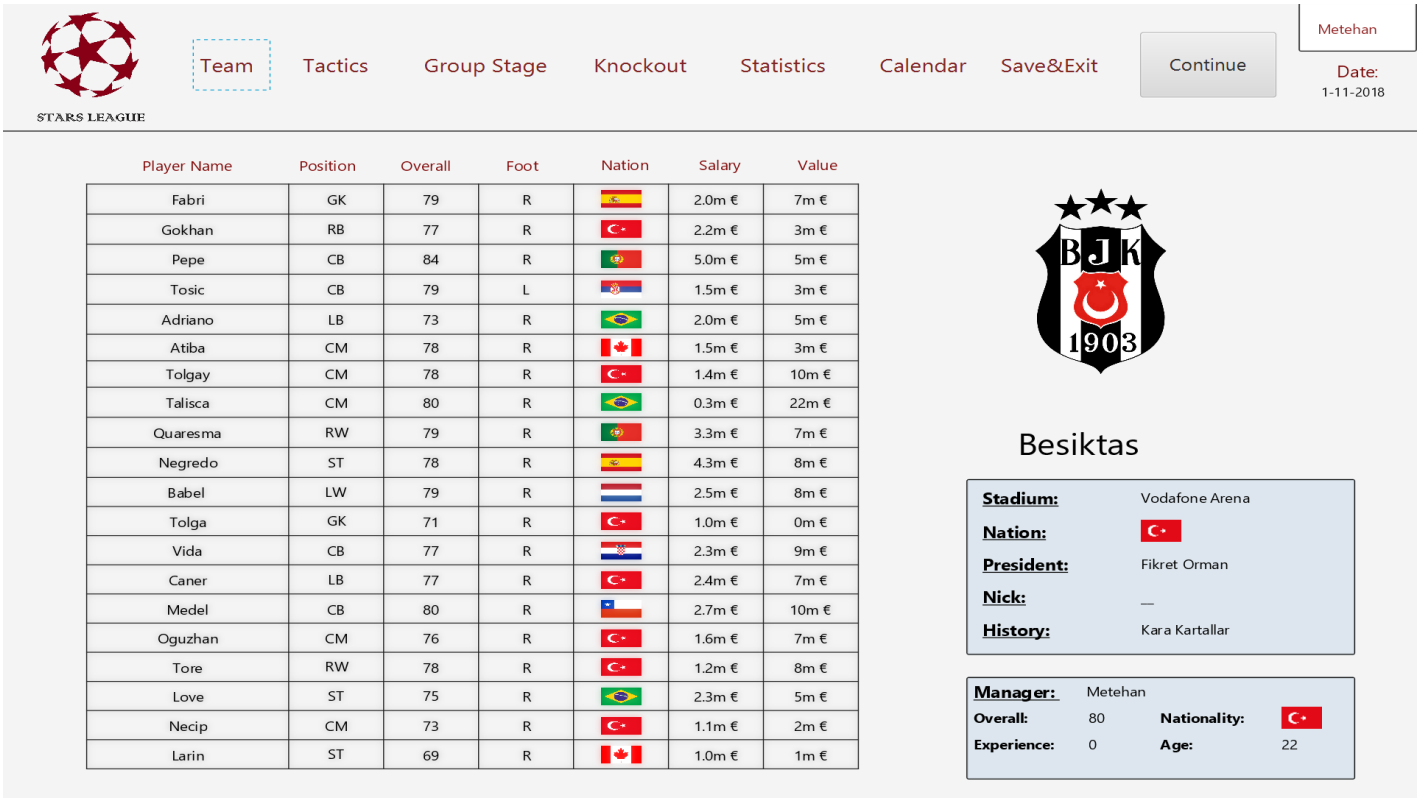

Figure 4.4.12 – Team View

This view includes information about team's stadium, nation, players, manager and president. If the user clicks on the name of a player figure 4.4.8 will be shown up.

5. Incomplete Features

In the second iteration of Analysis Report, we have mentioned that we will be adding Transfer feature to the "STARS League". On the other hand, in the meantime, to accomplish transfer feature we had to wait until most of the other feature function properly. Consequently, we had to postpone transfer feature until the end. Now, since we are out of time, we are unable to complete transfer feature of "STARS League".