



Bilkent University

Computer Science Department

Object Oriented Software Engineering Project

STARS League

Analysis Report

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

Course Instructor: Eray TÜZÜN

April 03, 2018

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

Contents

I Introduction

2 Overview

- 2.1 Gameplay
- 2.2 Roles
- 2.3 Statistics
- 2.4 Tournament

3 Requirements

- 3.1 Functional Requirements
 - 3.1.1 Play Match
 - 3.1.2 Choose Team
 - 3.1.3 Display Calendar
 - 3.1.4 Display Player
 - 3.1.5 Display Tournament Information
 - 3.1.6 Transferring a Player
 - 3.1.7 Listening Music
- 3.2 Non-Functional Requirements
 - 3.2.1 Ease of Use
 - 3.2.2 Performance
 - 3.2.3 Extensibility
 - 3.2.4 Pseudo Requirements
 - 3.2.5 Packaging
 - 3.2.6 Reliability

4 System Models

- 4.1 Use Case Models
- 4.2 Dynamic Models
- 4.3 Object and Class Model
 - 4.3.1 Tournament

- 4.3.2 Team
- 4.3.3 Elimination
- 4.3.4 KnockoutTree
- 4.3.5 Group
- 4.3.6 Match
- 4.3.7 Person
- 4.3.8 Manager
- 4.3.9 President
- 4.3.10 Player
- 4.3.11 Action
- 4.3.12 Goal
- 4.3.13 Injury
- 4.3.14 Card
- 4.3.15 YellowCard
- 4.3.16 RedCard
- 4.4 User Interface and Navigational Path
 - 4.4.1 Home Screen
 - 4.4.2 Help View
 - 4.4.3 NewGame View
 - 4.4.4 Team View
 - 4.4.5 Player View
 - 4.4.6 Tactic View
 - 4.4.7 Group View
 - 4.4.8 Knockout View
 - 4.4.9 Statistics View
 - 4.4.10 Calendar View
 - 4.4.11 Match Play View
 - 4.4.12 Navigational Path

5 Glossary Reference

I. Introduction

STARS League is a strategy game that uses the same logic as “Football Manager”. We have decided to implement such a game because our team is made of four football fans. We thought it would be more than fun to design game similar to “Football Manager” instead of just playing ready one. However, due to scarcity of time and resources we were forced to appease some of the most delicate parts of the game. Notwithstanding, we can add some of those functionalities throughout the semester as we advance with our project. For instance, after the first iteration we plan to add transfer feature. This feature would help the managers enhance their team. One more point we want to add, as of the first iteration, is different actions made by the players in the field. As of the time, players are able to score and make fouls that results red or yellow cards. These cards will penalty Later on we can give them ability to be injured and retreat from fields for a few matches. The user plays the game in the shoes of the manager of the team that he or she have chosen at the beginning of his or her career in the game. Manager’s career will continue until his/her team wins a cup or loses to a team and gets eliminated at some point during the league. The team will compete with 32 teams in total.

2. Overview

2.1 Gameplay

In “STARS League”, you must choose a team and continue with the team throughout the whole cup. User, in the position of manager of the team that he/she selects, can change his/her team only after the cup or when gets eliminated and prompted to restart. Afterwards, the manager decides the tactics of the team and team applies that tactic when they play against an opponent. Actions in the match will depend on the teams, players and other matches. Skills of the players, intelligence of the manager, past form of players will have an effect in a match. We will try to reflect real life situations in matches as much as possible in the “STARS League”. After each match, statistics will be updated accordingly. There will be competition to win the cup until the last match of the tournament.

2.2 Roles

In “STARS League”, there are different roles to represent a real life football tournament's people. Each football team will have its players, the manager and the president. Moreover, each game will have a referee. Detailed explanations of these roles are as following:

- **Player:** Person who is responsible for playing a match in a team. Each player has statistics such as goals scored, assists made and cards given accordingly.
- **Manager:** Person who is responsible for managing the team on the game field. He/She determines the transfers, tactics and starting roster.

- **President:** Person who is responsible for ruling the whole team. He/She makes economic and administrative decisions about the club.
- **Referee:** Person who makes decisions in the match such that giving cards.

2.3 Statistics

There is no doubt about the main component that makes the games so fun is statistics. So it is when implementing a game application. We rely on statistics to decide which team should win in a match, which player should score a goal and when. So that we will gather statistics for each player separately. We will do the same for teams in both group stage and knockout stage. We will decide which player is a top scorer or which player has the most assists in each stage by using the statistics of each player. Then we will have separate frame to display these tops.

2.4 Tournament

The tournament will have different stages. Teams will be in groups in the group stage and will try to beat each other to pass to the next round which is called knock-out stage. In the knockout stage, teams will face with each other to reach final match. The team who win the final match will become the champion of the tournament.

3. Requirements

3.1 Functional Requirements

3.1.1 Play Match

User have to play the next match of his/her team when a match day arrives. User need to organize his/her team before starting a match. Before the beginning of the match, there will be information about match including date, stadium, referee, weather of the match day, odds, results of previous matches of both teams, starting squads and reserve players on the screen. After match starts, user will be able to see the current minute and squads of the both teams. Also actions of the match will pop up on the screen while time passes.

Actions in the match depends on many factors. Firstly, like in the real life, home team -whose stadium is where the current match is played- is advantageous in many cases because they has most of the fans in the stadium.

Secondly, every team has different kind of players with different skills. These skills might be game-changer in the match. For instance player with good shooting ability is likely to score a goal in a game. Player with low mental ability are likely to get yellow and even red cards often in football matches. However also mentally good players might get a red card in the match. Also managers of the teams have different qualities therefore, user need to careful while playing against a team with a good manager.

Tactic, style and tempo are the main elements of the strategy of a team. In the “STARS League”, user will be able to determine the team’s tactic formation, playing style and playing tempo. These features will also have an effect in the match dynamically. User will be able to change these features while match is playing and before the match. For instance playing in a more attacking way may help the team to score goals however it might cause troubles in defense. Also putting a player into position that he is not suitable might lower the performance of the team. Thus, these features will have a great effect in the match actions and results.

Nonetheless, even all the factors point out the victory of the specific team, as in real life there is always a chance factor that might affect the game in different ways. For instance the team with worse players might get a win against the favorite team in the tournament. Therefore we will add that chance factor to make the game more interesting.

3.1.2 Choose Team

Before the tournament, the user has to make a choice from 32 given teams. Since quality of teams is important for the user, it should be allowed to give some general information including players, balance, etc. In other words, to make a good decision, displaying teams and accessing general information about them will be possible.

3.1.3 Display Calendar

A calendar of matches will be displayed. For each match, some simple information such as opponent team's name, logo, and home-away status will be visible. Also, this calendar will include days without matches. For the sake of progress, the user will be able to jump to next day. When the user prefers not to continue, it will be available to switch to other panels.

3.1.4 Display Player

STARS League provides wide information about players. For each player, general information including team, position, age, preferred foot, nationality, height, weight, value, wage, status will be visible. Also, attributes such as pace, shooting, dribbling, passing, defense, physical, goal keeping, mental are included. In addition to all of them, statistics including the number of matches played, goals, assists, yellow cards, red cards will be visible too.

3.1.5 Display Tournament Information

If the user wants to be the champion of the STARS League, it is necessary to pass 2 main steps of the game: Group stage and knockout stage. For group stage, general information about all groups, named from A to H, will be visible. It will be possible to see teams and points that they got, goals that they scored, wins, losses, draws, etc. The teams will be ordered according to their points, the results of the matches played between the teams that got same points, etc. For knockout stage, all

matches played up to the current time will be displayed. Eliminated teams will be colored with grey, while others are colored with red. For the sake of understanding, all names of the steps of the knockout stage, “Last 16”, “Quarter Final”, “Semi Final”, “Final”, will be visible.

3.1.6 Transferring a Player

Clubs can do transfers in the transfer window period for creating better teams before the STARS League. Thus, the possibility of success in STARS League will be increased if clubs doing good transfers. For doing transfers every club will have transfer budget which will be determined according to club values. Clubs can't spent more money than their transfer budget. If a player is send to the transfer list by his club, percentage of the accepted transfer offers will be increased. If a player is not on the transfer list, to get accepted transfer offer will be harder.

3.1.7 Listening Music

User will be able to play the song from the music list which is given by the game. User also able to mute the songs from the settings. Some musics can motivate the manager and provide to feel as a real manager. That would be increased the success rate at STARS League

3.2 Non-Functional Requirements

3.2.1 Ease of Use

The game is very easy to play in spite of its wide content. There is no need to use keyboard, the game is played with mouse only. The user can get information about the game, if he/she encounter with an issue during the game. Therefore, there must be an option to get information about the game widely. STARS League includes “How to Play?” option to provide information.

3.2.2 Performance

Performance is one of the important requirements. Response time is important from the user’s view. STARS League will have constant 60 FPS to make user more comfortable while playing the game.

3.2.3 Extensibility

Since extensibility and reusability are really important for software engineering, we decided to design the game in such a way that it can be modified later. This will lead to make STARS League more enjoyable. In the second iteration, we are planning to add new features which will increase satisfaction.

3.2.4 Pseudo Requirements

STARS League will be implemented in Java. The game is portable because it will run on JVM (Java Virtual Machine). Graphics will be implemented by using JavaFX and AWT libraries.

3.2.5 Packaging

The game does not require any installation because there will be a jar file to run the game. Just a double click on the jar file starts the game and opens the home screen.

3.2.6 Reliability

STARS League has Save Game feature for user to save his current progress in the tournament and continue with it later. There will be a autosave feature which user may want to select, which automatically saves the game after every match. Thus the progress of the user will be always safe with that feature. Otherwise, user need to explicitly save the game to protect his/her data.

4. System Models

4.1 Use Case Model

In this part includes information about the main use case model of STARS League. Use case detailed is on the below.

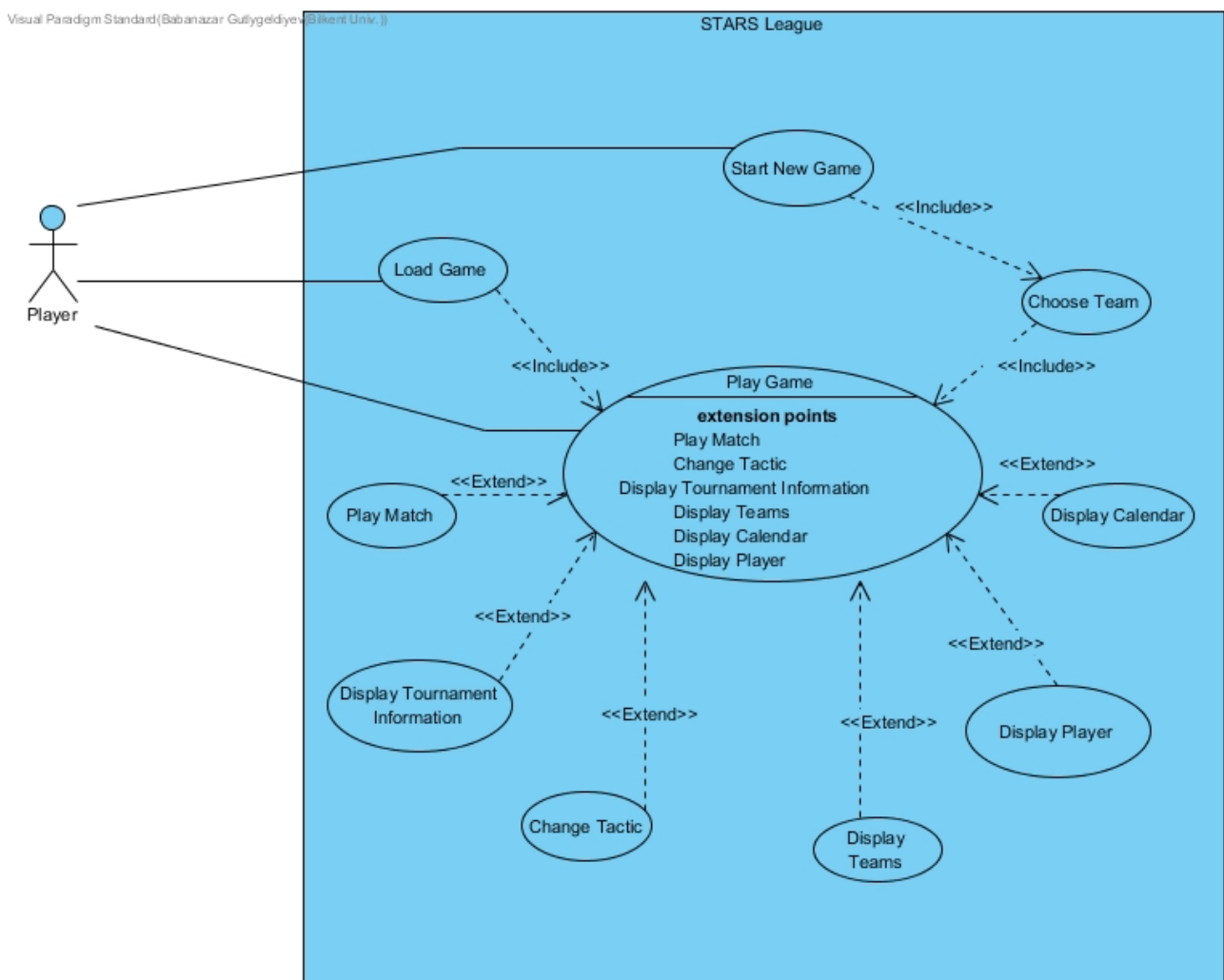


Figure 4.1 - Illustrates the use case model of Stars League

Use case #1

Use case name: Start New Game

Participating actors: User

Stakeholders and Interests:

- Player wants to start new game.
- System shows the team list to user for choosing the team.

Pre-conditions: The game should be opened from the player.

Post-condition: -

Entry condition: Player has already opened the game and is on main menu.

Exit condition: Player has chosen to quit the game via in-game menu.

Success Scenario Event Flow:

- System displays the team selection window to user.

Main Flow of Events:

1. Player starts a new game.
2. Player chooses a team to participate in tournament.
3. Player plays matches unless he is eliminated.

Alternative Flow of Event:

A. If player desires to exit from the game at any time:

A. Player selects exit button from the main menu.

B. System quits from the game and team selection window.

Use case #2

Use Case Name: Choose Team

Primary Actor: User

Stakeholders and Interests:

- Player chooses the team for participating the STARS League tournament.
- System keeps the team of the Player for participating the STARS
- League tournament.

Pre-condition: Player should press the “start new game” button to reach the team selection window.

Post-condition: The team which is chosen by player participating tournament and draws the groups according to this selection.

Success Scenario Event Flow:

1. The team is chosen by Player.
2. Chosen team is participated STARS League tournament.
3. Player can start the game by pressing “Start New Game” button after choosing a team.

Alternative Flows:

- A. If the player wants to choose new team:
 - A. Player presses “Main Menu” button to return to main menu and choose a new team.
 - B. System displays the Main Menu by starting new game.

Use case #3

Use Case Name: Play Game

Primary Actor: User

Stakeholders and Interests:

- Player starts to manage the team which is chosen by him/her.
- System starts the game.

Pre-condition: For starting to game, user has to press “Start New Game” button on the main menu and choose the team for determining which team would be managed by player.

Post-condition: -

Entry Condition: Player chooses the team for starting the game. **Exit**

Condition: Player selects “Return to Main Menu” button or “Exit” button from bar.

Success Scenario Event Flow:

1. Player chooses the team for starting to game.
2. Game is started by the System.
3. Group stage of STARS League tournament would be drawn by the system randomly.
4. Matches of group stages and knock-out stage will be scheduled by the system.
5. System allows player to manage its own team.
6. Player changes tactics of his/her own team.

Alternate Flows: -

Use case #4

Use Case Name: Load Game

Primary Actor: Player

Stakeholders and Interests:

- Player wants to continue the game from the last save date.
- System allows player to continue his/her game from the last save date.

Pre-Condition: Player has to be in Main Menu of the STARS League.

Post-Condition: The loaded document would be changed after user press the save button which is on the Main Menu.

Entry Condition: Player selects “Load Game” button from the main menu.

Exit Condition: Player selects “Back” button from return main menu.

Success Scenario Event Flow:

1. Player presses “Load game” button to continue the game from the last saved file.
2. Game displayed the user according to last saved file.

Alternative Flows:

- A. If Player wants to return to the main menu at any time:
 - A. Player selects “Return to Main Menu” button to return the main menu instead of loading the game.
 - B. System displays the Main Menu instead of loading the game.

Use case #5

Use Case Name: Play Match

Primary Actor: Player

Stakeholders and Interests:

- Player wants to play a match after changing his/her own team tactic to qualify from the group stage and knock-out stage of STARS League tournament.
- System simulates the matches according to player's ratings and display the live match to player.

Pre-condition: Player has to start the game after choosing a team to play game with his/her team.

Post-condition: Match simulation has been played by the system and displayed the result.

Entry Condition: Player selects "Play Game" button to play game.

Exit Condition: Player selects "Main Menu" or "Exit" button to exit from the match.

Success Scenario Event Flow:

1. Player presses "Play Match" button to start the match.
2. Live center of the match screen displayed by the system
3. According the result of the match, the game would update the group stage standings and fixtures successfully.

Alternative Flows

- A. Player selects "Main Menu" button to exit match before it is finished.
- B. Player selects "Tactics", "Fixtures", "Standings" buttons to examine the situation of opponents and his/her own team.

Use case #6

Use Case Name: Display Tournament Information

Primary Actor: Player

Stakeholders and Interests:

- Player wants to see the standings of the teams.
- Player wants to see the statistics of the players and the teams.
- System displays a list of players and teams' statistics.
- System displays the standings of the teams.

Pre-conditions: Player should play game to access display tournament information.

Post-condition: -

Entry Condition: Player selects "Display Tournament Information" button from the bar.

Exit Condition: Player selects "Back" to return previous screen.

Success Scenario Event Flow:

1. Player selects "Display Tournament Information" from bar.
2. System displays the standings or knockout stage tree of the STARS League tournament.
3. System displays the statistics of the teams and players.

Alternative Flows: -

Use case #7

Use Case Name: Change Tactic

Primary Actor: Player

Stakeholders and Interests:

- Player wants to change the tactic of the team which is managing by him/her.
- System displays the tactic screen and player list to the player for changing tactic.

Pre-condition: Player has to choose a team for starting the game. Thus, he/she can access the tactic button to change tactics of his/her own team.

Post-condition: Player selects “Back” to return previous screen.

Entry Condition: Player selects “Change Tactic” button.

Exit Condition: Player selects “Back” to return menu.

Success Scenario Event Flow:

1. Player presses “Change tactic” button to make changes on tactics on managing team.
2. Tactic screen is displayed to Player in “Change Tactic” button by the System.
3. Player changes the tactic of the managing team according to it
4. is own favorite tactic.
5. System updates the new tactic successfully.

Alternative Flows:

- A. If Player wants to use default tactic for the managing team.
- B. Player selects “Default Tactic” button from “Change Tactic” screen.
- C. Default tactic is updated by System.

- D. If Player requests to return previous menu at any time.
 - a. Player selects “Back” button from the bar.
 - b. Player returns the previous screen.

Use case #8

Use Case Name: Display teams

Primary Actor: Player

Stakeholders and Interests:

- Player desires to view the player list of the managing team, for checking the player ratings, values, positions, etc.
- System displays the player list of the managing team

Pre-condition: Player has to choose a team for starting the game. Thus, he/she can access the display the tactic button to display player list of his/her own team.

Post-condition: Player list would be ordered by player’s request.

Entry Condition: Player selects “Display Teams” button.

Exit Condition: Player selects “Back” to return previous screen.

Success Scenario Event Flow:

1. Player presses “Display Teams” button to display the player list of the team.
2. Player list of the managing team is displayed to player.
3. Player orders the list of the team player list.
4. System updates team list ordering successfully.

Alternative Flows: -

Use case #9

Use Case Name: Display Player

Primary Actor: User

Stakeholders and Interests:

- Player desires to see the statistics of the player, information of the player, rating of the player and attributes of the player by display player.
- System displays the player information, statistics, rating and attributes of the player.

Pre-condition: Player has to choose a team for starting the game.

Thus, he/she can access the player display screen to be get information about players which are players of managing team of the player.

Post-condition: -

Entry Condition: Player selects “player name” from the team list. **Exit**

Condition: Player selects “Back” to return previous screen. **Success**

Scenario Event Flow:

1. Player presses the player name to display the player.
2. Rating of the player displayed by the System.
3. Attributes of the player displayed by the System.
4. Information of the player displayed by the System.

Alternative Flows: -

Use case #10

Use Case Name: Display Calendar

Primary Actor: User

Stakeholders and Interests:

- Player wants to see the calendar to see the match dates for doing preparation for incoming matches.
- System display the calendar to the player with match dates.

Pre-condition: Player has to choose a team for starting the game. Thus, player can see the incoming matches' dates for doing preparations to his/her managing team.

Post-condition: Schedule is displayed by the system.

Entry Condition: Player selects "display calendar" button from the bar.

Exit Condition: Player selects "Back" button to display the previous screen.

Success Scenario Event Flow:

1. Player presses "Display Calendar" button for displaying the schedule.
2. System displays the calendar to the player.
3. Player sees the incoming matches.
4. Player checks the calendar for preparations for incoming matches.

Alternative Flows: -

4.2 Dynamic Models

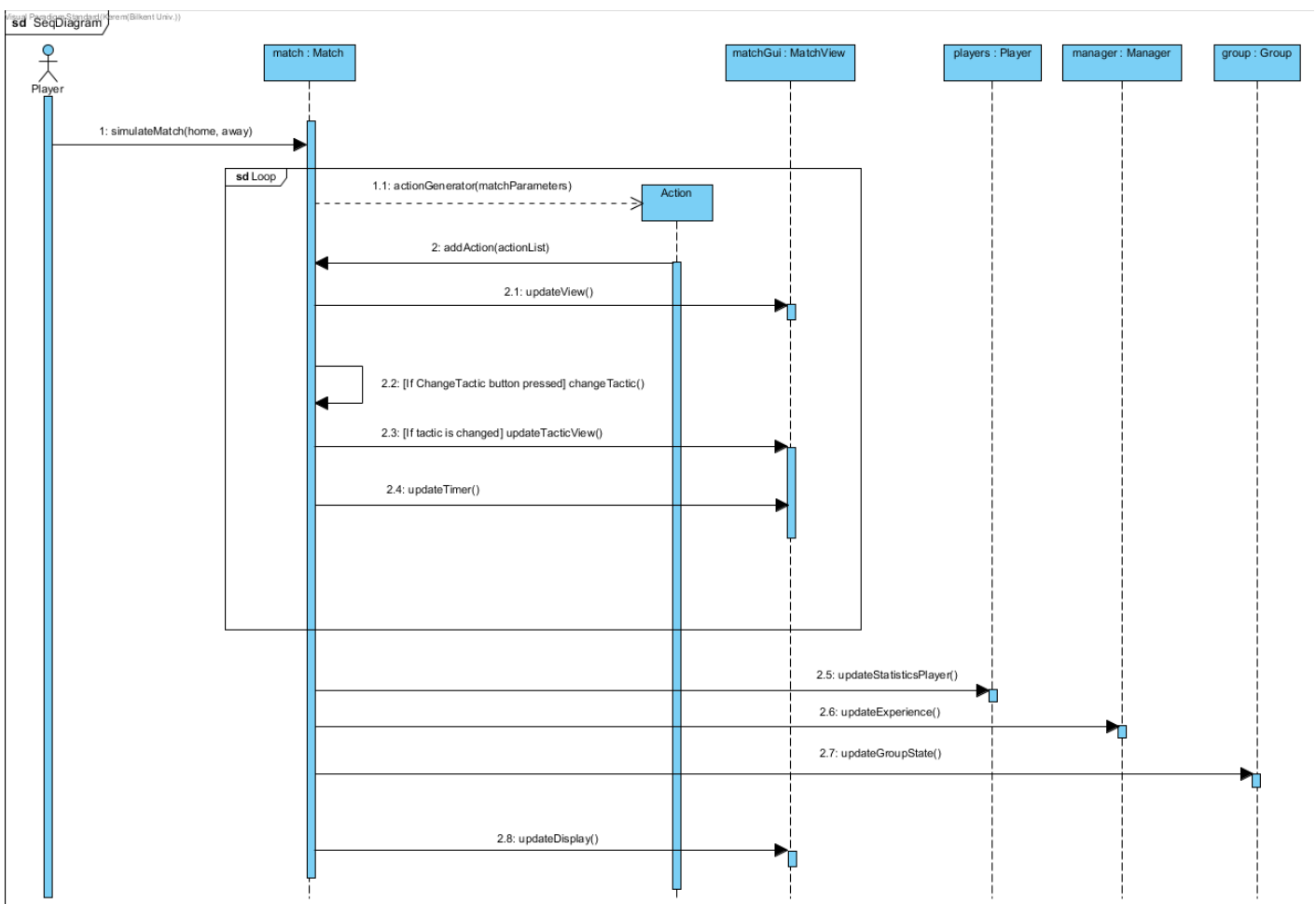
4.2.1 Sequence Diagram

We introduce the most important sequences in our sequence diagrams. One of them is Play Match, which represents the event flow and object communication while player wants the next match in his/her schedule.

1. Play Match

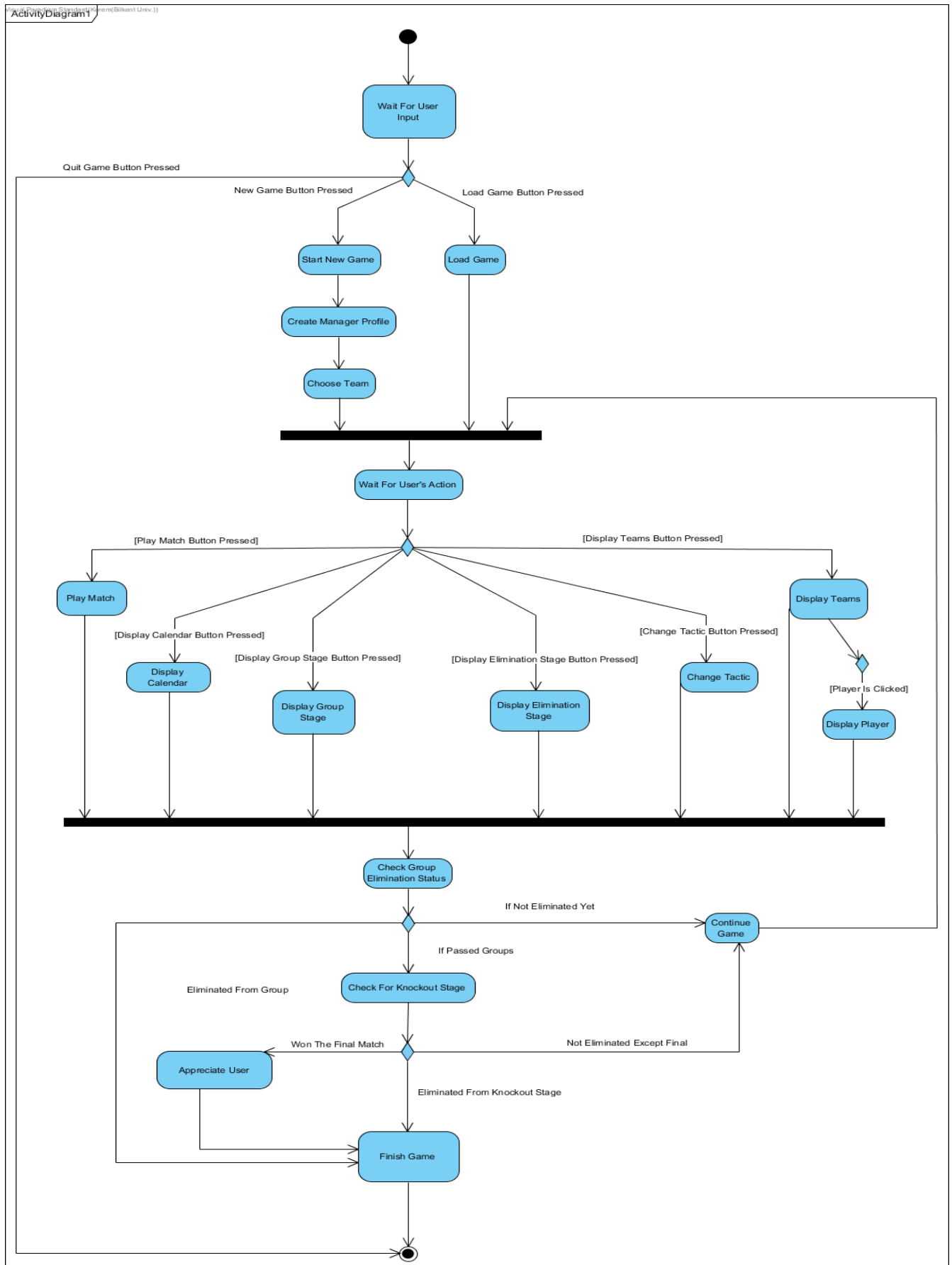
Scenario Name: Play Match

Scenario:



Description: During the match, many actions are created randomly. However, players' attributes and tactic also have some effects on these actions. After creation of an action, according to its type, there may be some changes related to player, manager. Also, for each action, its type and time is displayed on the screen.

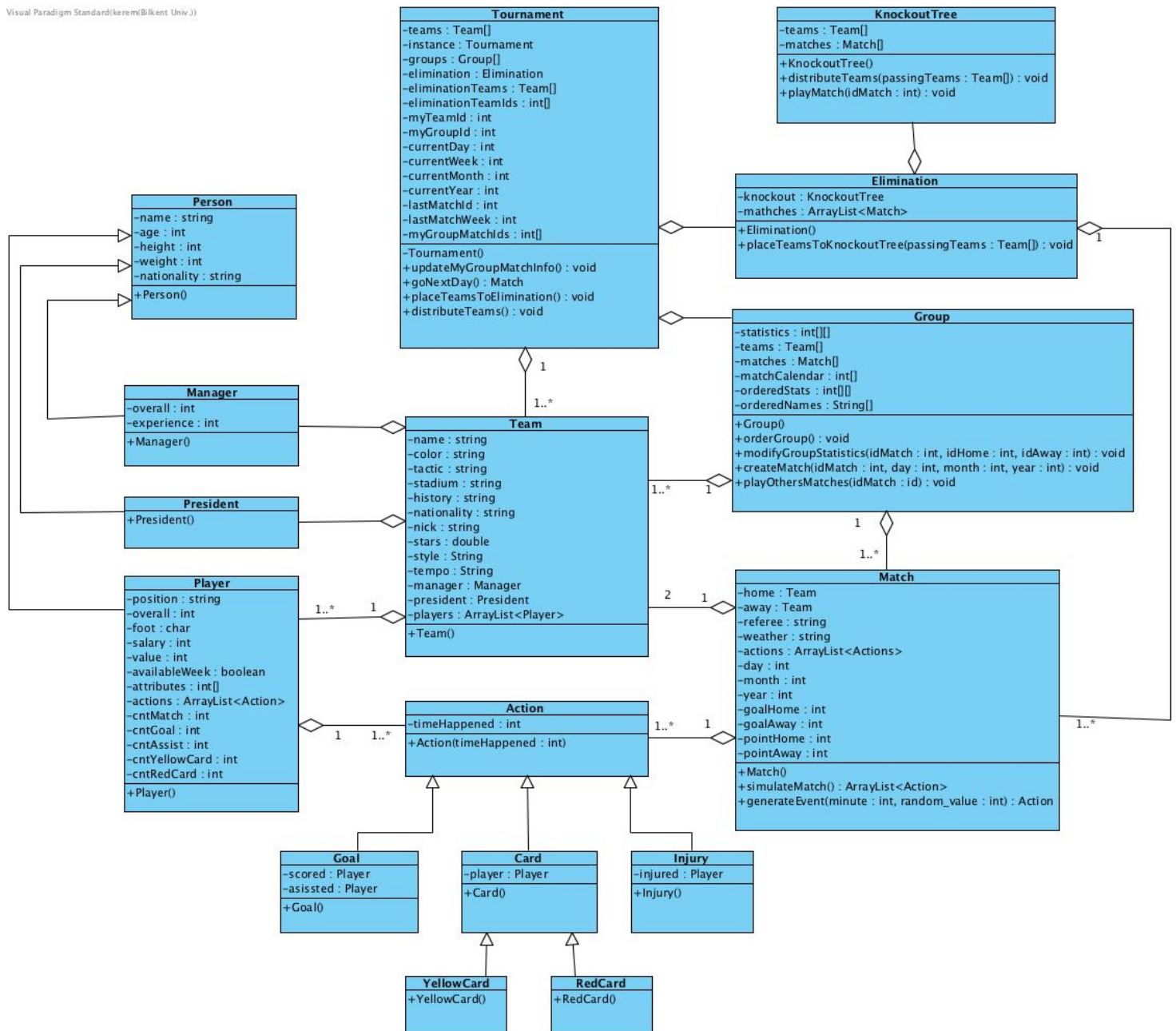
4.2.2 Activity Diagram



Description: Activity diagram shows the main flow of gameplay. At the beginning of the game, it checks whether new game or load game is chosen. Then, the system controls which tab is clicked. There are many options for the user. These are “Play Match”, “Team”, “Group Stage”, “Knockout”, “Player Statistics”, “Calendar”, “Save&Exit”.

4.3 Object and Class Model

Visual Paradigm Standard(kerem@ilkenet Univ.)



4.3.1 Tournament

Tournament is the main model class of the project. It is the center of flow mechanism.

Methods:

+ `updateMyGroupMatchInfo()`: After a match of user is played, updates the data related to number of total goals, assists, cards, and modifies group table.

+ `goNextDay()`: If user has a match on that day, he plays the match. Otherwise, time shifts by one day. Also, other teams play match if there is any.

+ `placeTeamsToElimination()`: After group matches finished, puts top 16 teams into elimination stage.

+ `distributeTeams()`: Before the group stage, shuffles the teams and put them into 8 groups where there are 4 teams for each group.

Attributes:

- `eliminationTeams`: Stores 16 teams that passed group stage successfully.

- `lastMatchId`: Used to update the statistics which is related to the latest match.

- `myGroupMatchIds`: Stores 6 matches that user plays during group stage.

4.3.2 Team

Team stores the players, manager, and president, including general information of it.

Attributes:

- tactic: Can affect the results of matches. “4-3-3”, “4-4-2”, “4-2-3-1” are different examples to tactic that indicates how the players are distributed to different roles.
- style: Can affect the results of matches. “Attack”, “Defensive”, “Holding” are different examples to style that indicates how the players will act during the match.
- tempo: Can affect the results of matches. “Fast”, “Normal”, “Slow” are different examples to tempo. It affects the energy level of players.

4.3.3 Elimination

Elimination stores information related to elimination stage, including matches.

Attributes:

- knockout: Stores knockout tree that is related to status of the teams in the elimination stage.
- matches: Stores all matches played up to now.

4.3.4 KnockoutTree

KnockoutTree is main point of the elimination stage. It stores information related to the status of the teams in the stage.

Methods:

- + distributeTeams(): Shuffles the teams which passed group stage, and matches each corresponding pair which will play matches with each other.
- + playMatch(): Plays the given match. If it is final round, winner of the match becomes the champion. If it is the second match that is played between the given teams, results of both matches decide which of them will pass the round.

Attributes:

- matches: All matches of the elimination stage. Includes the matches which are not played yet.

4.3.5 Group

Group stores information related to first stage, group stage, of the tournament. This information includes statistics, teams, matches played between the teams of the group, dates of the matches.

Methods:

- + orderGroup(): Order the teams of the group according to total points that they collected during the group stage up to now.
- + modifyGroupStatistics(): Looks all matches played up to now for given group, and modifies statistics of the teams.
- + createMatch(): Creates matches of the group by using dates of them which are prepared before.

+ playsOtherMatches(): Plays matches which are between the teams that the user did not select at the beginning of the game. It is required, because they are not simulated and there is no delay between the actions of them.

Attributes:

- statistics: Stores 7 statistics which are number of matches played up to now, total wins, total draws, total losses, total scored goals, total conceded goals, total points for each team of the group.
- orderedStats: Shows statistics of the teams when they are sorted according to the total points that they collected.
- matchCalendar: Stores date of the 12 matches of the group.

4.3.6 Match

Match is one of the important classes for the flow the game. A sequence of matches decides the champion of the game.

Methods:

- + simulateMatch(): For each moment of the match, by using the attributes of players, tactics of teams, and manager, it decided whether there is going to be a type of action or not for that minute.
- + generateEvent(): Generates action and handles special cases related if there is any. Red cards and injuries results in special cases. For instance, a player who got red card at 61' cannot score a goal at '74.

Attributes:

- actions: Stores all actions happened during the match.

4.3.7 Person

There are 3 types of it, which are manager, player, and president.

4.3.8 Manager

Manager has an effect on the team that he manages, and also matches that he attends.

4.3.9 President

President of the team.

4.3.10 Player

Stores general information and statistics of the player which are modified during the tournament.

Attributes:

- availableWeek: Shows if the player can play a match at the current week. Injuries can affect this status.
- actions: Stores all actions that player has effects on.

4.3.11 Action

Action can occur during the match. There are 3 general types of it, which are goal, card, injury.

4.3.12 Goal

A goal has two players related to that, who are the player who scored the goal, and the player who assisted.

4.3.13 Injury

A player can be injured during the game. This injury may result in absence for several next matches.

4.3.14 Card

There are two types of card, which are yellow card and red card.

4.3.15 YellowCard

A player can get yellow card if he does not act too dangerously. An additional yellow card given to the same player results in a red card. There is only one player related to a red card.

4.3.16 RedCard

If a player gets a red card, he cannot play further for next several matches. There is only one player related to a red card.

4.4 User Interface and Navigational Path

4.4.1 Home Screen

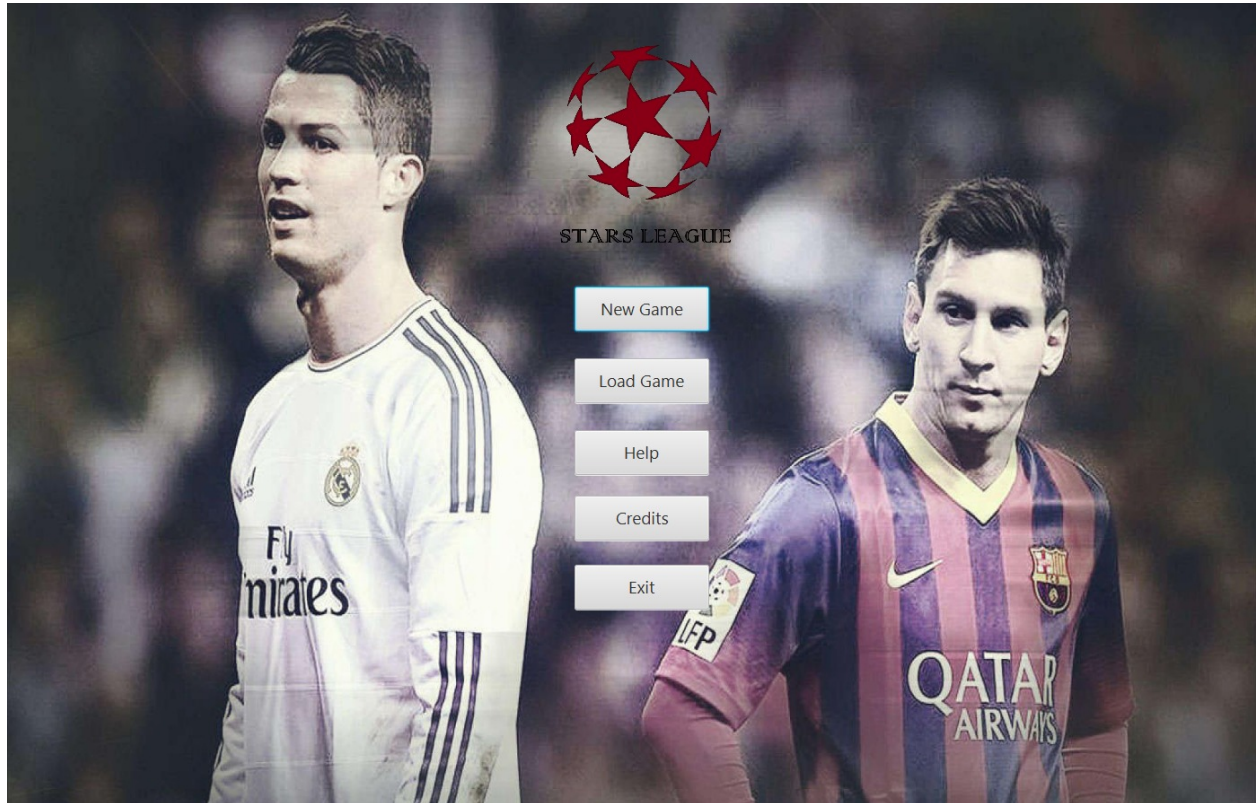


Figure 1 - Home Screen View

This is the screen shown to the user when she or he opens STARS League at first. From this screen, the user can proceed with “New Game” button to create new game or “Load Game” according to his/her desire. However, if the user clicks “Load Game” before any game is saved then it will not do anything. In other words, Load Game button will not react if there is no any saved game in STARS League’s database. Moreover, this screen contains buttons that lead to Help of the game and Credits of it, respectively.

4.4.2 Help View

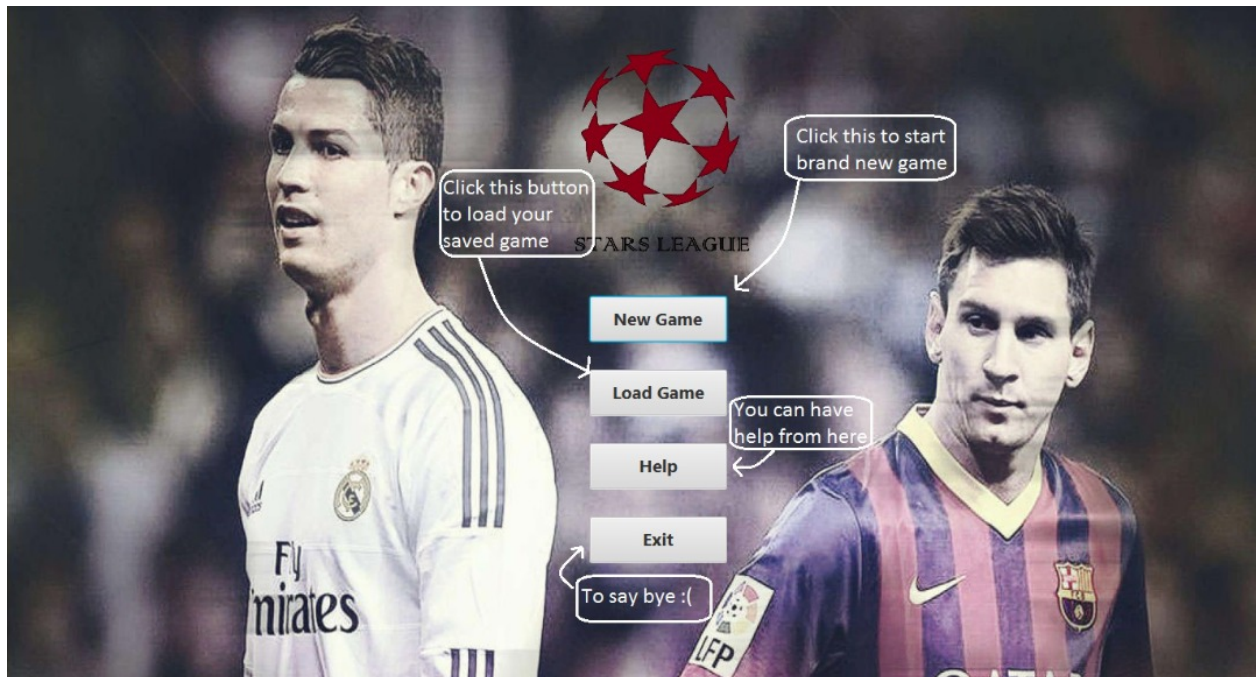


Figure 1 - Help view

Help view contains the information to start your career in STARS League. The new users can press to this button in order to require help about the STARS League.

4.4.3 New Game Screen

Afterwards, whenever the user clicks New Game button, STARS League will proceed to profile creation view. The user will enter the name, age, height, weight and nationality to the required fields accordingly. After that when Continue button is pressed STARS League will go on to the team selection part of new game creation.

1. Create Your Profile

Name:

Age:

Height:

Weight:

Nationality:

Figure 3 - Profile Creation View

Team select view will have names of 32 teams. The user will be prompted to select one of these teams as his or her team. He or she will be the new manager of the team of choice. The user will continue with his or her preferred team throughout the league of STARS. Before the user makes his or her choice there is no team selected. Without selecting any team, the user is will not be allowed to proceed to next stage. After selecting the team, the user will need to press Done button to go on.


2. Choose Your Team

- ☐ AFC Ajax
- ☐ AS Monaco
- ☐ FC Barcelona
- ☒ Galatasaray
- ☐ Paris Saint Germain
- ☐ Manchester United
- ☐ Chelsea
- ☐ FC Porto
- ☐ AS Roma
- ☐ Olympique Lyonnais
- ☐ Bayer Leverkusen
- ☐ RB Leipzig
- ☐ SL Benfica
- ☐ AC Milan
- ☐ FC Basel
- ☐ Spartak Moscow
- ☐ Sporting Lisbon
- ☐ Liverpool
- ☐ Real Madrid
- ☐ Borussia Dortmund
- ☐ CSKA Moscow
- ☐ Besiktas
- ☐ Manchester City
- ☐ Napoli
- ☐ Zenit
- ☐ Bayern Munchen
- ☐ Atletico Madrid
- ☐ Inter Milan
- ☐ Fenerbahce
- ☐ Tottenham Hotspur F.C.
- ☐ Juventus
- ☐ PSV Eindhoven

Figure 4 - Team Select View

4.4.4 Team View

After the user clicks Done button, the next screen that follows Team Select View is Team View. This is the screen the user would see if he or she had a saved game and pressed the Load Game button. This is the point the game starts and the league is formed. In this team view screen, the user sees all information about the chosen team. All his or her team's players are displayed in this screen. The user will find other details like stadium, history, etc of the team. Here Stars is the the stars team has earned throughout its lifetime. Moreover, s/he will be able to see information about the players when clicked on players. It is enough to click on any attribute of the player that the user wants to see information about. For instance, let's assume that the user clicked on the name Gomis.



Team

Tactics

Group Stage

Knockout

















Statistics


Calendar

Save&Exit

Continue

STARS LEAGUE

Player Name	Position	Overall	Foot	Nation	Salary	Value
Muslera	GK	83	R		83	10m €
Mariano	RB	80	R		82	8m €
Maicon	CB	82	R		83	14m €
Serdar	CB	79	R		81	9m €
Nagatomo	LB	78	R		80	6m €
Selçuk	CM	79	R		78	5m €
Fernando	CM	82	R		84	9m €
Belhanda	CM	80	R		80	12m €
Feghouli	RW	81	R		82	14m €
Gomis	ST	83	R		84	10m €
Garry	LW	79	R		83	16m €
Carrosco	GK	78	R		70	3m €
Linnes	RB	77	R		75	5m €
Ahmet	CB	92	R		73	2m €
Denayer	CB	96	R		78	5m €
Latovlevici	LB	98	R		70	1m €
Donk	CM	97	R		76	1m €
Eren	ST	93	R		77	3m €
Sinan	RW	98	R		77	6m €
Yasin	LW	91	R		78	4m €



Galatasaray

Stadium:

Ali Sami Yen

History:

Cimbom

Nation:

Turkey

Nick:

GAL

Stars:

5.0

Figure 5 - Team View

4.4.5 Player View

In this screen, the user will find information about the player that was clicked on. The user will find every information starting from personal information such as age, height, foot and position. Other statistics like professional statistics and attributes will be placed in this screen. Here the status of the player indicates the availability of the players for the next match. It becomes “unavailable” if the player is injured or gets the red card in the previous match. Moreover the user will see overall ranking of the player at the top right of the screen.

General

Team:	Galatasaray	Height:	180cm
Position:	ST	Weight:	70kg
Age:	30	Value:	10.0m €
Foot:	R	Salary:	3.4E-5m €
Nation:		Status:	Available

Attributes

Pace:	95	Defence:	93
Shooting:	92	Physical:	100
Dribbling:	93	Gk:	99
Passing:	93	Mental:	98

Statistics

Match Played	Goals	Assists	Yellow Card	Red Card
12	0	0	0	0

Figure 6 - Player View

4.4.6 Tactic View

On the top of the screen, the user finds Top Menu to choose among team, tactics, group stage, knockout stage, statistics and calendar. He/She will have option to save and exit the game if she or he prefers. If the user clicks Tactics button he or she will see accurate view that

presents tactics of the team. At the right side, by clicking combo buttons, he can change his team's players according to his/her preference. When he wants to change positions of the players of starting lineup, he can just do it with the similar way that is done for other players. Only difference is that two players just exchange positions. As seen, there are 11 options when the user wants to change positions of the players one for each position in the team. This means that only one goalkeeper can be assigned to the GK position. Moreover, on the right side of the screen, user can make choice between the tactics, styles and tempo of his/her team.

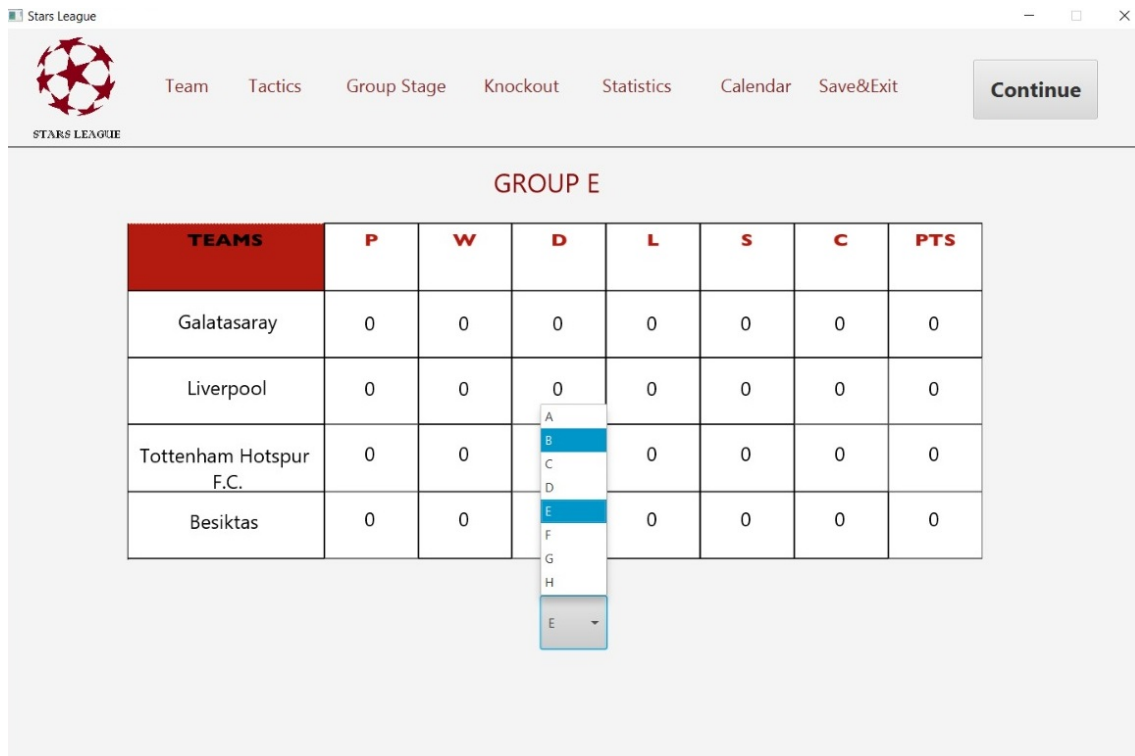


Figure 7 - Tactic View

4.4.7 Group View

Hereafter the user can see the progress she/he making at the league. For that he needs to press Group Stage button. When Group Stage button is pressed Group Stage View is brought to front. At first Group A will be displayed. However, the user can make preference of his team's

group by using the combo button at the bottom of the page. In this page, P is for played matches of the team among the teams in the same group. W, D and L represents the number of wins, draws and loses, respectively. S and C are for scored and conceded goals in the matched played within these teams. PTS is abbreviation of the points of each team in every group.



TEAMS	P	W	D	L	S	C	PTS
Galatasaray	0	0	0	0	0	0	0
Liverpool	0	0	0	0	0	0	0
Tottenham Hotspur F.C.	0	0	0	0	0	0	0
Besiktas	0	0	0	0	0	0	0

Figure 8 - Group View

4.4.8 Knockout View

After the group phase is completed, knockout stage will be formed. This knockout will be displayed in the figure 9. Winner of the ever pair of the teams will be marked red. On the other hand, the loser team will marked with faded grey color. At the end, the winner of teams on the right and winner of the teams on the left side will play final match for the STARS Cup.

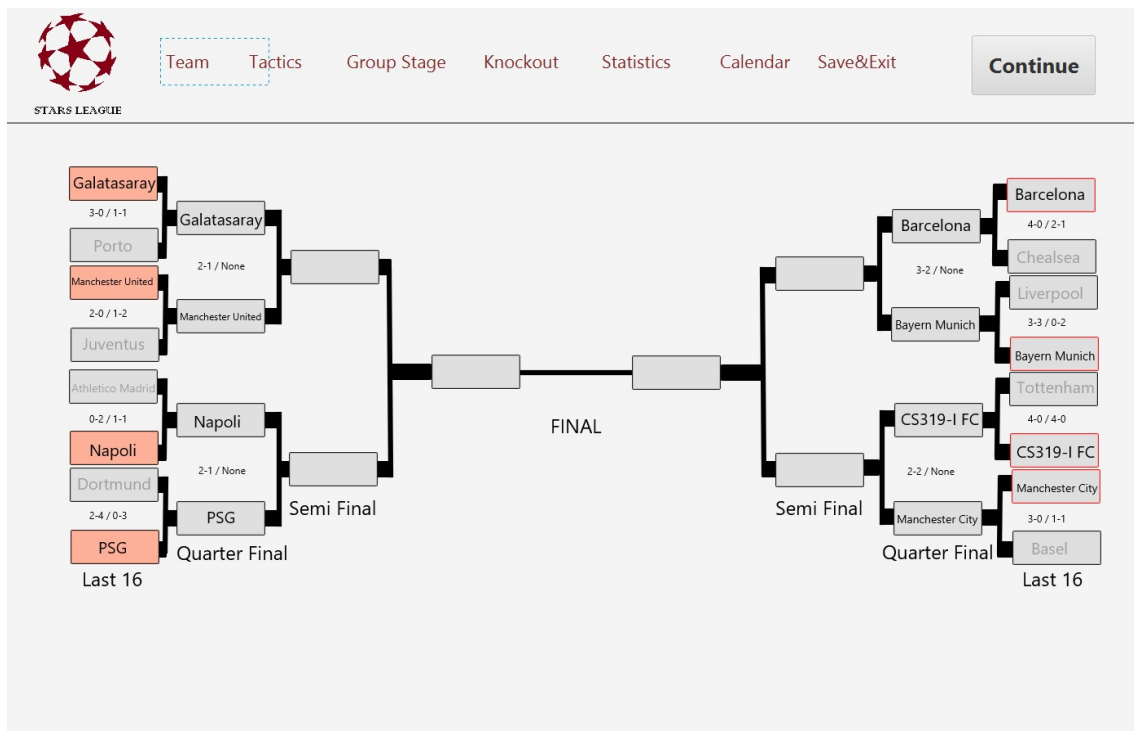


Figure 9 - Knockout View

4.4.9 Statistics View

Statistics View will present information about the tops of the league i. e. Stars of the Tournament. Here the user will see the names of the players with top scores top assists, the most number of yellow cards and the most number of red cards. These names will be updated after each match.

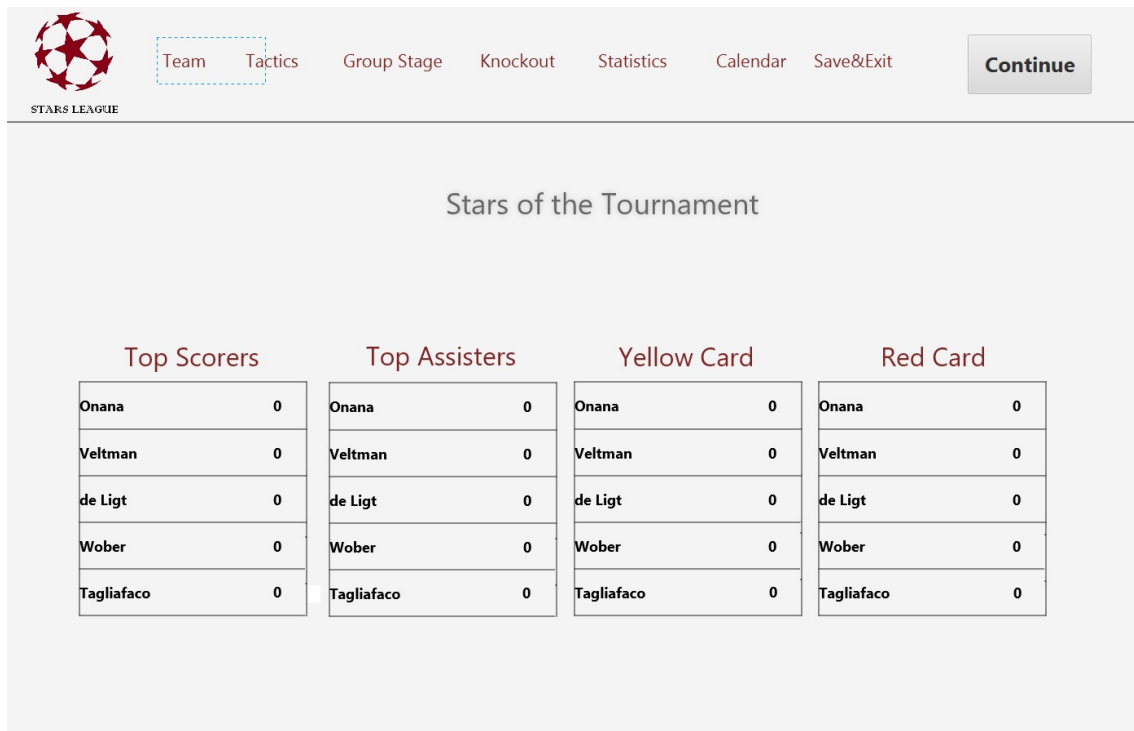


Figure 10 - Statistics View

4.4.10 Calendar View

Next to the Statistics button, the user will see Calendar button which contains the calendar of the tournament. This view will display the matches the user's team will play in the related days. Moreover it will mark the current day on the calendar. As time proceeds in the STARS League, events in the calendar will proceed accordingly. Played match days will display the scores of the matches on that day's spot.

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
01 Kas	02 Kas	03 Kas	04 Kas	05 Kas	06 Kas	07 Kas
08 Kas	09 Kas	10 Kas	11 Kas	12 Kas	13 Kas	14 Kas
15 Kas	16 Kas	17 Kas	18 Kas	19 Kas	20 Kas	21 Kas
22 Kas	23 Kas	24 Kas	25 Kas	26 Kas	27 Kas	28 Kas
29 Kas	30 Kas	01 Ara	02 Ara	03 Ara	04 Ara	05 Ara
06 Ara	07 Ara	08 Ara	09 Ara	10 Ara	11 Ara	12 Ara
13 Ara	14 Ara	15 Ara	16 Ara	17 Ara	18 Ara	19 Ara
20 Ara	21 Ara	22 Ara	23 Ara	24 Ara	25 Ara	26 Ara

Figure 11 - Calendar View

4.4.11 Match Play View

In the long run, the user will forward the time by pressing Continue button at the most right. When Continue button is pressed match play screen will be prompted. This screen will display every information about the match and the playing teams. On two sides tactics and lineups with logos of both teams are placed. In the top middle, score of the match will be displayed. The screen underneath will display actions accordingly in neat format. At the bottom, Pause and Play buttons are placed to pause and play again the match, respectively. In between these two buttons, timer displays the time of the match. The match will end when the timer hits 90.

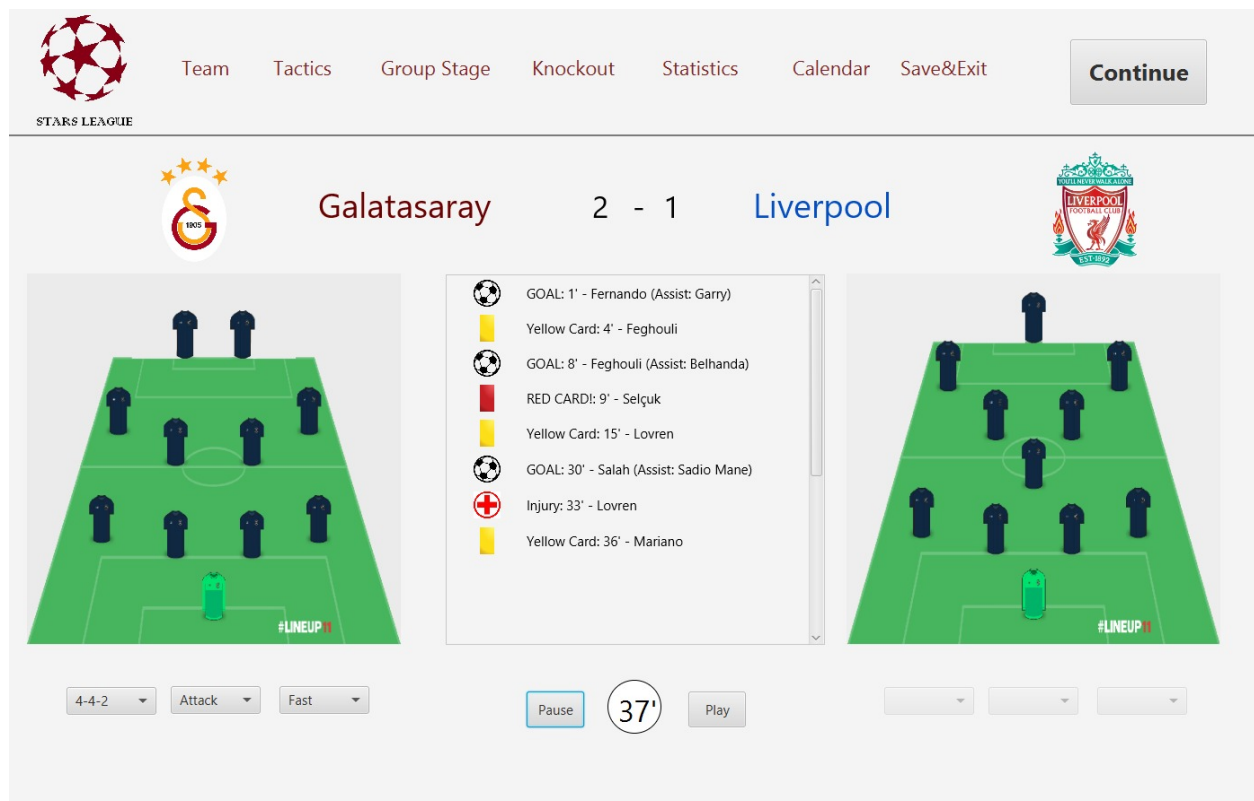
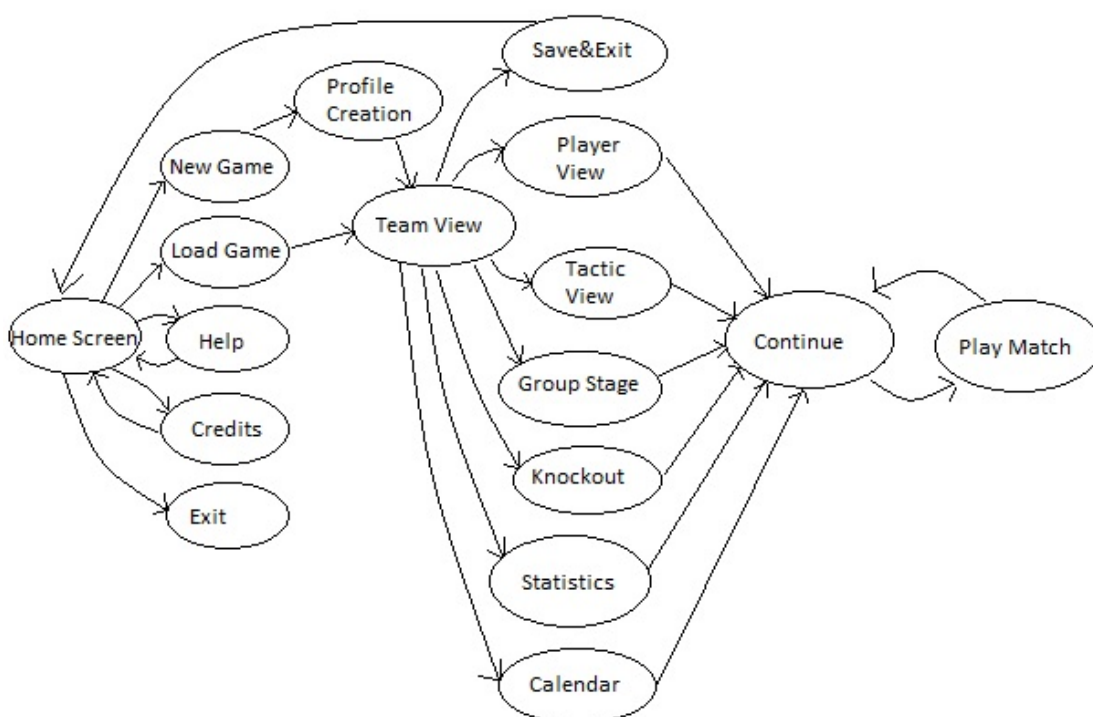


Figure 12 - Play Match View

4.4.12 Navigational Path



5. Glossary and References

1. B. Bruegge and A. H. Dutoit, *Object-Oriented Software Engineering, Using UML, Patterns, and Java*, 3Rd Edition. Prentice-Hall.
2. “Balsamiq,” *Balsamiq*. Balsamiq Studios, LLC , 16-Oct-2017.
3. C. H. K. Tsang, “Object-oriented technology from diagram to code with visual paradigm for UML.” .