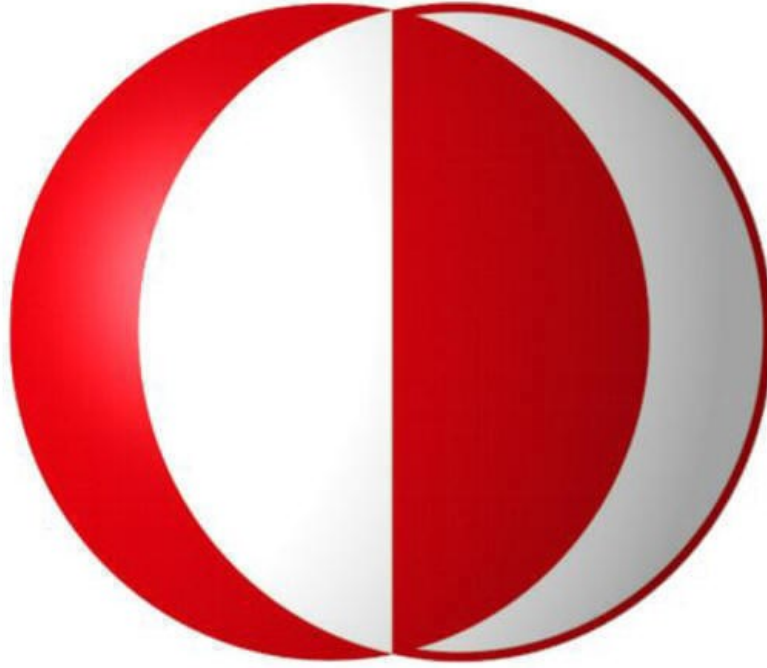


Software Requirements Specification

30.10.2013

FootboFun Social Network Game



The Originals

Ahmet Abak e1630490

Can Gülaydın e1560226

Çağrı Utku Akpak e1745660

Doğancan Arabacı e1818921

İbrahim Arslan e1745751

Table of Contents

1 INTRODUCTION	3
1.1 PROJECT DEFINITION	3
1.2 PURPOSE	3
1.3 SCOPE	3
1.4 DEFINITIONS ACRONYMS AND ABBREVIATIONS	4
1.5 REFERENCES	4
1.6 OVERVIEW	4
2 Overall Description.....	5
2.1 Product Perspective	5
2.1.1 System interfaces.....	5
2.1.2 User interfaces	5
2.1.3 Hardware interfaces.....	5
2.1.4 Software interfaces.....	5
2.1.5 Communication Interfaces	6
2.1.6 Memory.....	6
2.1.7 Operations	6
2.1.8 Site adaptation requirements	6
2.2 Product Functions	6
2.3 Constraints	7
2.4 Assumptions and Dependencies	8
3. Specific Requirements.....	10
3.1 Interface Requirements	10
3.2 Functional Requirements.....	10
3.2.1 User Functions	10
3.2.1.2 Managing Living Area.....	10
3.2.2 Admin Functions	14
3.2.3 Data Analyst Functions	14
3.3 Nonfunctional Requirements.....	14
3.3.1 Performance Requirements	14
3.3.2 Design Constraints	15
4 Data Model and Description	16
4.1 Data Description	16
4.1.1 Data objects	16
4.1.2 Data Dictionary	20
5 Behavioral Model and Description.....	21

5.1 Description for Software Behavior..... 21

 5.1.1 Events..... 21

 5.1.2 States..... 22

5.2 State Transition Diagrams..... 24

6 Planning..... 25

 6.1 Team Structure 25

 6.2 Estimation 25

 6.3 Process Model..... 26

7 Conclusion..... 26

1 INTRODUCTION

This section introduces the requirement specification document for the Footbofun Social Network Game. It provides the purpose and scope of the project. Any definitions and references are listed in this section as well as an overview of the remaining requirements specification document.

1.1 PROJECT DEFINITION

This project is a social networking game, which will simulate football career, social life and assets of a young football player. There is no visual match simulation but it will be possible to track the football match from commentator's output. The aim of the users is to be successful in football career, social life and assets of simulated life. It is possible to train the character, attend nightlife, acquire a girlfriend, buy houses and cars, football and gym equipment, sign for sponsorship agreements, transfer to other teams, etc. The users of the game interact with each other in order to compete. Users are able to join the game only with their Facebook accounts.

1.2 PURPOSE

The purpose of this document is to give a detailed description of the requirements for the "FOOTBOFUN" software. It illustrates the purpose and complete declaration for the development of system. It also explains system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for both the development team to develop the first version of the system and the test team to test the software appropriately.

1.3 SCOPE

The "FOOTBOFUN" is an event-based social network game. It should be targeted the people who is interested in football and who like football simulation games. The application of the project should be mobile game application which is free to download from a mobile phone application store (ios, android) and should be available on the Internet via "facebook" social networking site.

Furthermore, the project is going to be a client server architecture multiplayer social game with a data mining module. All system information is maintained in two databases: NoSQL database namely Couchbase which will be used for storage of the user data and persistent game data which is located on a web-server, the second database is a MySQL database which will store the all actions of the user for data mining purposes. Therefore; users need both the Internet connection on their devices that runs this software and a facebook account to be able to join the game.

1.4 DEFINITIONS ACRONYMS AND ABBREVIATIONS

- IEEE: Institute of Electrical and Electronics Engineers
- iOS: Apple Mobile Operating System
- Android: Google Mobile Operating System
- TCP: Transmission Control Protocol
- IP: Internet Protocol
- HTTP: Hypertext Transfer Protocol
- MySQL: My Structured Query Language
- NoSQL: Not Structured Query Language
- Json: Javascript Object Notation
- IDE: Integrated Development Environment

1.5 REFERENCES

- IEEE. IEEE Std 829-2008 Standard for Software and System Test Documentation. IEEE Computer Society, 2008
- IEEE. IEEE Std 830-1998 Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998
- IEEE. IEEE Std 1016-2009 Standard for Information Technology -System Design-Software Design Descriptions IEEE Computer Society, 2009

1.6 OVERVIEW

This document provides a high-level description of the “FOOTBOFUN” multiplayer social game. The document then describes general software and hardware constraints as well as any assumption and dependencies concerning the projects. The majority of this document focuses on the specific requirements list. A master list of specific requirements is given first, followed by each requirement explained in detail in the next section. Both the internal and the external interfaces are addressed in the subsequent section. The internal interface requirements are requirements involving system interfaces and the external interface requirements are requirements involving user hardware, software and communications interfaces. This requirements document concludes with conclusion section of the whole document.

2 Overall Description

2.1 Product Perspective

(FUTBOFUN) is a multiplayer social network game, which simulates football career, social life and assets of a young football player. There is no visual match simulation but it is possible to track the football match from commentator's output. An event-based simulation which is generated by statistical parameters of teams and players is used. The purpose of the game is to be successful in football career, social life and assets of simulated life. Moreover, this project is going to support cross-platforms, i.e. iOS, Android and Web using the same server. In development process, Java and Flash technologies, native iOS and Android platforms will be used. All in all, those who will use this product will interact each other with different platform.

2.1.1 System interfaces

- TCP/IP Socket Protocols: This interface will be used for data communication between server and clients.
- Touch Screen Interface: Running on mobile devices, this interfaces captures touch events and sends it to the application.
- Mouse Interface: Running on PC devices, this interfaces captures mouse events and sends it to the application.

2.1.2 User interfaces

The project interfaces are developed for 3 different platforms, namely iOS, Android and Web. On Android and iOS platforms the product is a native application and on Web the application runs on a flash player supported web browsers. All platforms have similar user interfaces but different screen layouts to fit user interface on different devices. All of those user interfaces contain six menus for six main functionality of this product. These functionalities are explained briefly in the 2.2.

User who must use the system can easily understand the user interface, since it is a game and has a short tutorial which will be provided to the user at the beginning of the game.

2.1.3 Hardware interfaces

Since the game must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable. Moreover, this project needs a mobile device or PCs to run. The server side of the product should constantly run on a cloud server which is part of a Server PC.

2.1.4 Software interfaces

1. Flash Player, version 10 or later, is required for web client to run on browsers.
2. iOS, version 5.0 or later, is required for iOS mobile devices to run natively.
3. Android, version 2 or later, is required for Android mobile devices to run error-free.
4. Java Runtime Environment (Java RE), version 1.6 or later, is required for java server which is used on the server side of the product. Therefore, all operating systems can be used for running the server of the product.
5. Apache Mina Networking Framework, version 2.0.7, will be used for the TCP/IP socket connection handling part of the server.

6. Apache HTTP Server, version 2.4 or later, will be used as http server for web client.
7. Couchbase Database, version 2.2.0, will be used as the main database of the server. Couchbase client libraries is going to be used by the java server for database communications.
8. MySQL, version 5.6, is going to be used for database on data mining module of the product.
9. Google Json Library (Gson), version 2.2.4 or later, will be used to serialize data objects for communications and database records.
10. Facebook iOS, Android and Web Graph APIs, version 3 or later, will be used for Facebook connection on different platform respectively.
11. Native iOS Frameworks, currently stable versions, are required for iOS client side part.
12. Android Java Libraries, currently stable versions, are required for Android client side part.

2.1.5 Communication Interfaces

The server and client side of the game will use the HTTP protocol for communication over the internet and for the intranet communication will be through TCP/IP protocol suite.

2.1.6 Memory

The memory requirement of the mobile clients is at least 256MB primary memory.

2.1.7 Operations

1. When the number of users playing the game increases, the number of java server can be increased.
2. When the number of users playing the game increases, the number of Couchbase modules can be increase to meet the users' needs.
3. For protection of the user data, the company can back-up the data stored on Couchbase database without stopping its operation.

2.1.8 Site adaptation requirements

- The Couchbase and MySql databases should be installed and operational on the cloud units of the server. For couchbase database, at least one data bucket needs to be created.
- Java Runtime Environment should be installed and operational on the cloud units of the server.
- A site domain is required for web client of the product.

2.2 Product Functions

This product will provide these features to all who join in the game. User will interact these features on product user interface;

1. User login via Facebook

Facebook login is mandatory so user must login via Facebook to reach the game environment. Facebook login system will be used for system registration because it provides a robust security layer and it can be used for secure login and sharing. In addition to this, Facebook is widely used all around the world and it has social network features which can be integrated to different products.

2. Social life management

At this part of the product, user will be able to manage social life of own football player. To illustrate social life, it has many parts such as; hanging out with a girlfriend, buying a gift to her, taking care of his family, going out with teammates.

3. The Manager and Finance

The manager makes life of the football player easy. At this menu, user can search for a team, apply to a team, make/refuse sponsorship agreements, and make/refuse investments.

4. Football Club

Football player will make training, speak with the club manager and choose his playing style while in his football club.

5. Living Area

User can rest, have party, make self-training, also can buy home, car and other stuffs which are provided by the game.

6. Player Profile

User will be able to change own player's style. User also can buy new items and take energy drink. In addition to this, user can see statistics of the player from this part.

7. News (Notification Center)

At this part, user will see the news about his own player's events. For example, when a player is asked for sponsorship and transfer offer, this news will appear on this notification center also known as news. Moreover, when player makes some charity events, notification will also appear on this part.

2.3 Constraints

1. Frameworks

Facebook Web, iOS and Android, Apache Mina frameworks knowledge is mandatory.

2. Memory Management

Development for mobile platforms is needed to care about memory issues because of limited memory.

3. Operating Systems

For iOS development, developer must work on OS X operating system.

4. Hardware Limitation

Since the project has limited hardware support for server, programmer must be care of CPU and Memory usage.

5. Security and Reliability

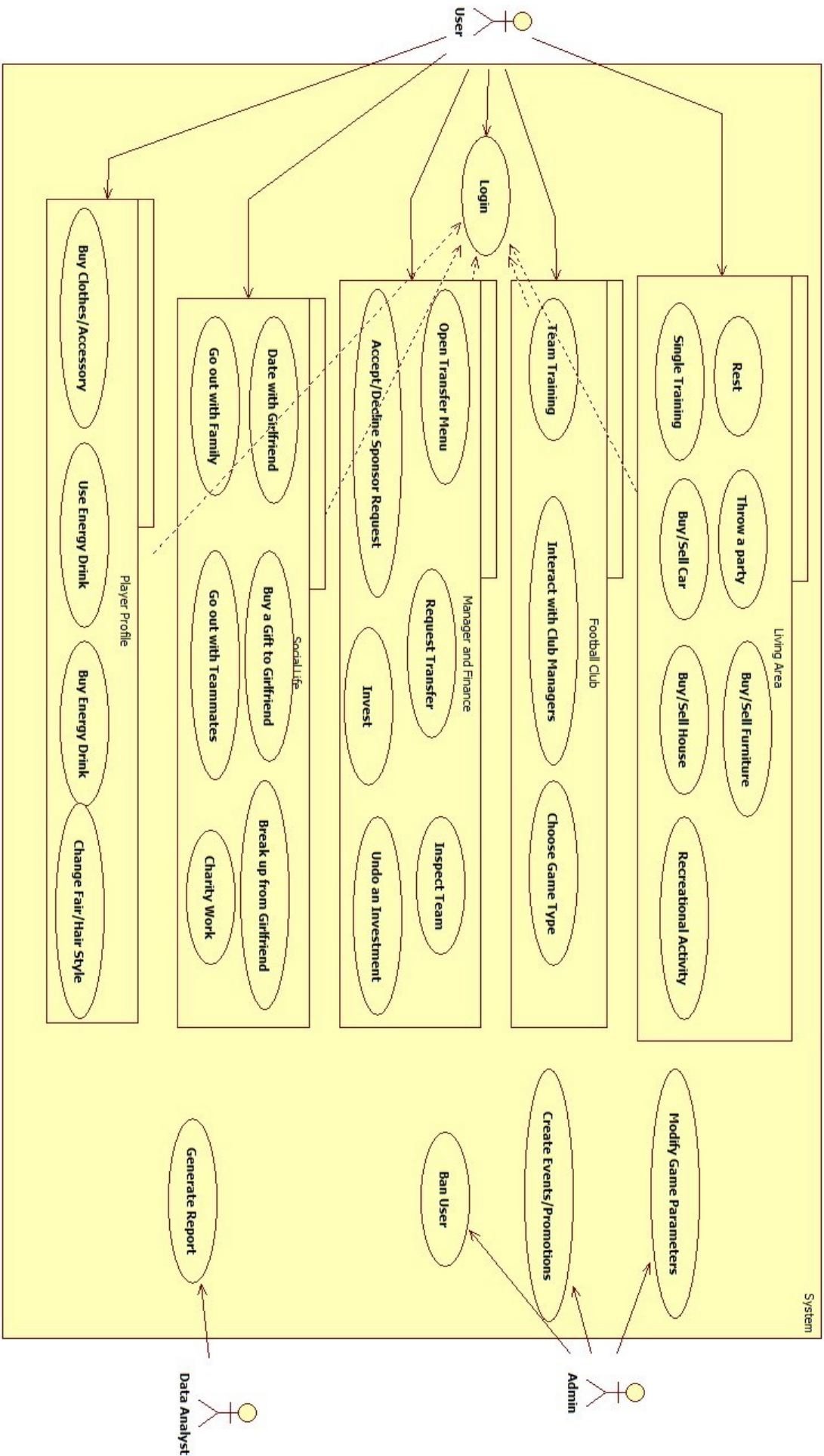
The security of the user registration, login authentication and store transactions should be provided by Facebook security system. The reliability of data communication should be provided by TCP/IP.

6. Programming Languages

Java, action-script and objective-c knowledge is mandatory.

2.4 Assumptions and Dependencies

This project is a server-client application and its web client side will be developed on Adobe Flash. Currently, all web browser which shows the web client support flash. Moreover, since this project is a game project and it contains many features, some of those features may be popular so other features which are like popular features can be added to the project in the future for more attractiveness and those additions will affect server side, database and client side interfaces.



3. Specific Requirements

3.1 Interface Requirements

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla or Netscape Navigator by which user can access to the system. And also application should be reached from mobile phones with iOS and Android system.

The interfaces in iOS and Android should be similar and mobile graphics should be designed same for both of them.

All interfaces' functionalities should be same and explained in functional requirements section.

3.2 Functional Requirements

The functional requirements are the requirements of what the system should do

3.2.1 User Functions

User functions section explains what a user can do in the game.

3.2.1.1 Login

When user starts the application on a mobile device, he should see login screen first. Login screen should clarify that user can only login via Facebook. Once user logged in with his Facebook account, it should be saved on device and after that login screen won't be available again.

For a user, who reaches the application on a web browser should launch the game from Facebook only. There won't be any login screen, since users are already logged in.

3.2.1.2 Managing Living Area

There are 6 main screens (Football player's life's main parts to manage.) in our game.

In living area section user shall see his ambit, house and car. And also he shall interact with those.

3.2.1.2.1 Rest

The character in the game should have an energy attribute, which goes down after matches, trainings and some events such as, going out at night. From house image, he shall let rest his character.

Depending on confort level of the house, the character should rest faster or slower.

3.2.1.2.2 Have a Party

With interacting the house, a user shall throw a party in his house. Parties have a regeneration time, which is maximum 18 hours. He can reduce this time by some products from shop.

After the party, user's morale should be higher but energy should be lower.

If user has no girlfriend during this party, he can meet one.

3.2.1.2.3 Single Training

If the user has necessary equipment in his house, he shall train himself. Training should reduce his energy but also increase his stamina level.

Single trainings won't be effective as much as team trainings.

Single training should have a cooldown time, which is maximum 3 hours. He can reduce this time by some products from shop.

3.2.1.2.4 Buy/Sell Car

When user clicks/touches his car in front of his house, he should be redirected to car shop. In car shop he can buy a new car.

Buying a better car should increase his morale.

The user can have only one car at a time. If he buys a new one, the old car should be automatically sold.

3.2.1.2.5 Recreational Activity

Recreation activities includes watching TV, reading a book, playing console games etc.

When user choose to spend recreation time, his morale should be higher. His energy won't be effected. Recreation time has a cooldown time, which is maximum 3 hours. He can reduce this time by some products from shop.

3.2.1.2.6 Buy/Sell Furniture

When user interacts with his furniture part, he should be redirected to furniture shop. In there, he can buy a new furniture set.

Buying a better furniture set should increase effect of party, rest and training events.

The user can have only one furniture set at a time. If he buys a new one, the old set should be automatically sold.

3.2.1.2.7 Buy/Sell House

The user shall buy a new house any time. He can be redirected to house shop from living area screen. New house can result change in ambit change. There are 3 available ambits:Suburbs, Skyscrapers and Mansions.

A better house should increase his morale regeneration and he should have more furniture set options.

The user can have only one house at a time. If he buys a new one, the old house should be automatically sold.

3.2.1.3 Managing Social Life

In social life section, user should see relation status of his family, his friends from team and his girlfriend if he has one. Moreover, he'll have a chance to go out or make charity work from this screen.

3.2.1.3.1 Date With Girlfriend

If he has a girlfriend, he shall go out with his girlfriend. After he chooses to date with her, he'll be able to choose which place to go. Going better places should increase his morale and relationship status with his girlfriend.

There should be a cooldown time, which is maximum 3 hours. He can reduce this time by some products from shop.

3.2.1.3.2 Buy a Gift to Girlfriend

User can buy a gift to his girlfriend at any time. Depending on value of the gift, his relation status with his girlfriend should be increased. If he buys a cheap gift, the girlfriend may be upset, which results to reduce of relation status.

3.2.1.3.3 Break Up With Girlfriend

When you have a girlfriend, it's impossible to meet a new one. That's why when user decides to change his girlfriend, first he should break up with his old one.

Breaking up with girlfriend should decrease player's morale.

3.2.1.3.4 Go Out With Family

By interacting with family, the user can choose to go out with his family. This event should increase both player's morale and his relation status with his family.

Going to more expensive places should effect increase amount. He can meet a new girlfriend in such events.

3.2.1.3.5 Go Out With Team Friends

By interacting with family, the user can choose to go out with his team friends. This event should increase both player's morale and his relation status with his team friends.

A better relationship with his teammates should result in more chance to get pass from them.

Going to more expensive places should effect increase amount. He can meet a new girlfriend in such events.

3.2.1.3.6 Charity Work

From socail life screen, the user should be available to make some charity works.

Attending to charity works, should increase morale and fans of the player. He can meet a new girlfriend in such events.

3.2.1.4 Managing Club Life

In club life screen, user should see information about his team. All events, related to team should be handled in this screen.

3.2.1.4.1 Team Training

The user is able make one training match in a day. This match should affect his chances to be in line up.

Training should decrease energy of the player.

After training he may gain some skill points.

3.2.1.4.2 Interact With Club Managers

User shall interact with club manager. He shall ask for better facilities and a better contract. There should be a cool down time, which is maximum 3 hours. He can reduce this time by some products from shop.

Club administartion should answer the player immediately.

3.2.1.4.3 Choose Game Type

The user should have the opportunity to choose his game style. He can choose between defensive/offensive, passive/aggressive, selfish/sharing and hard-working/lazy.

Every game style should affect matches differently. It should be possible to read explanations of game styles in this screen.

3.2.1.5 Manager Room Functions

In manager room, the user should see his own manager. He can have managers with different levels and abilities. Better managers can find better teams or do better investments.

In this room, the user can manage sponsors, investments and transfer options.

3.2.1.5.1 Open Transfer Menu

The user shall see transfer menu from this scene. In transfer menu, he can do following: Inspect a team, request for transfer.

3.2.1.5.2 Inspect Team

The user shall search for all teams in his world. He can see empty slots of a team, team level and basic information of the related team.

3.2.1.5.3 Request Transfer

While inspecting teams, if the users thinks that he can transfer to a team; he'll be able to make a transfer request to that team. If his manager is good enough, he should receive a contract from that team in same day.

3.2.1.5.4 Accept/Decline Sponsor Proposals

In manager room scene, he can see contracts from possible sponsorship agreements. He can read conditions of the contract and decide to accept or decline the contract.

3.2.1.5.5 Invest

The player's manager should manage his investments. The player should see possible and ongoing investments belong to him. On possible investments, there should be a risk possibility value, which shows possible success rate of the investment. Based on this, user can choose to invest or not.

Investments' success rate also depend on player's manager's skills. A better manager should result in better investments and more profit.

3.2.1.5.6 Undo an Investment

The player should see list of his current investments. From this list he can undo his investment. After that, the investment won't be visible to player anymore.

3.2.1.6 Manage Character

In this section, the player should see his own character, his statistics, skill points and his trophies.

3.2.1.6.1 Buy Clothes/Accessory

By interacting character's clothes/accessory he shall go to clothes/accessory shop. Here can buy new items, which increase his morale and fans.

3.2.1.6.2 Buy Energy Drink

The user should see amount of energy drinks, he owns. If he wants to buy more, he can go to shop and get more energy drink.

Energy drinks are used to increase energy amount of the player.

3.2.1.6.3 Use Energy Drink

The user should see his energy level and amount of energy drinks in character section. He can choose to use one energy drink to increase his energy.

3.2.1.6.4 Change Face Style/Hair Style

The user shall go to hairdresser and change his hair and beard.

Changing his style to a better style should result in morale and fan increase.

3.2.2 Admin Functions

Admin functions section explains what an admin can do in the game system.

3.2.2.1 Add/Remove Events

Admins can directly interact with server and add/remove events. This gives the administrators the ability to create custom event and promotions. For instance, the administrator can create a limited size tournament or lower the price of an object for a limited period while the game is active. The increases the administrator control over the game and also increases the interaction inside the game.

3.2.2.2 Change Game Parameters

This action gives data analyst ability to generate dynamic reports based on their specifications. These reports in going to be generated using all the automatically stored actions of all the users playing the game. The accuracy and information content of reports based on the number of players using the game.

3.2.2.3 Ban User

If an admin is sure that a user cheats, he should be able to ban him with his unique ID.

3.2.3 Data Analyst Functions

Data analyst functions section explains what a data analyst can do in the game system.

3.2.3.1 Generate Data Mining Report

This action gives data analyst ability to generate dynamic reports based on their specifications. These reports in going to be generated using all the automatically stored actions of all the users playing the game. The accuracy and information content of reports based on the number of players using the game.

3.3 Nonfunctional Requirements

3.3.1 Performance Requirements

In our system, there should be parallel worlds which run on different servers. In this section performance requirements of one single world should be given.

In a world there should be 64 countries. In each country there should be 8 different leagues. (1 Super league, 1 First League, 2 Second Leagues, 4 Amateur Leagues) Each league should have 16 teams. As

a result there should be 128 different teams in one single country. In each team there are bots and there may be users. Maximum possible user amount per team is 10. All in all there can be 81920 unique users per world. (64 Countries * 8 Leagues * 16 Teams * 10 Players).

81920 threads in same time is the worst case of the system for a single server, but it's not likely to become a real issue. We expect users to be connected 3 on one day on average. As one day we take 16 hours of a day as possible hours for players to be online. This yields us to the result, which is there should be 15360 sessions per hour. As conclusion, the system should handle 256 different sessions in 1 minute.

3.3.2 Design Constraints

The system should be written in Java language and the server shall use TCP/IP for communicating with clients. The server shall use Apache Mina frameworks. There should be three types of client side, which are web browser, iOS and Android.

For web browser client side, Flash shall be used. The application should be available in Facebook. For Android client side, Java with Android Development Kit shall be used. For iOS client side, Objective C on XCode IDE shall be used. All necessary native libraries shall be used.

Since the provided hardware for server is limited, developers should care about performance and memory issues.

System should be scalable. I.e. the system should be designed such that, to create a new world with all 64 different countries in the game, using a new server should be enough. All parallel worlds should be different from each other.

Users shouldn't be able to reach databases or modify databases. On client side, there shouldn't be any critical data about the user.

4 Data Model and Description

This section describes information domain for the software.

4.1 Data Description

Data objects that will be managed/manipulated by the software are described in this section.

4.1.1 Data objects

We can categorize the data object of our product into four different categories. The first category is associated with user information that is sent to the clients and stored on the server. There are two important classes that shows the foundation of user information.

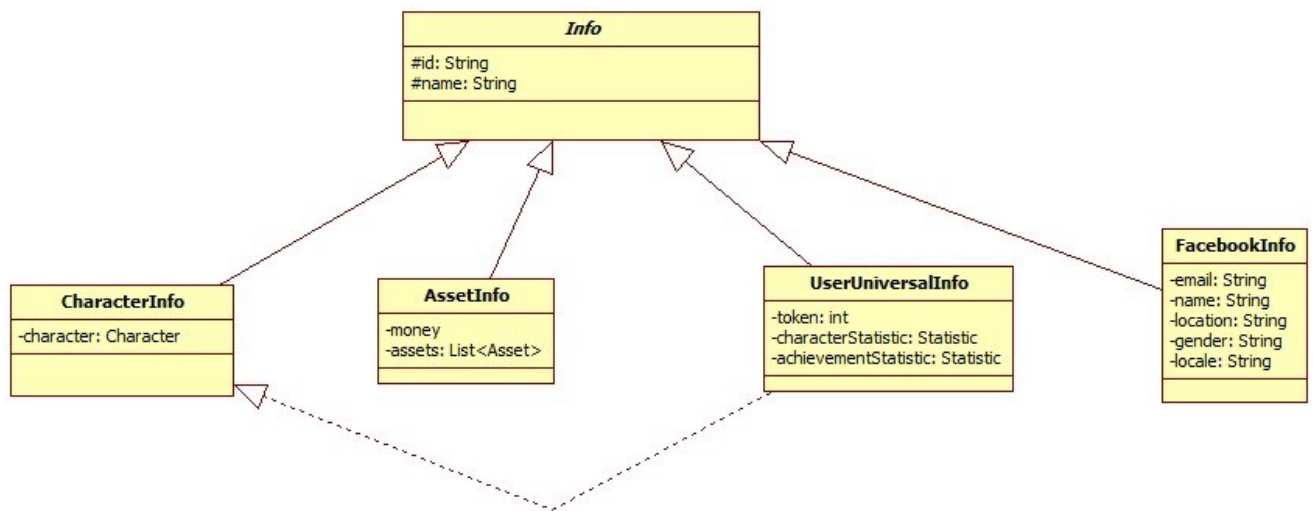
User Class



User class contains the most basic information about a user. Specific information about the game is stored inside the subclasses of the info class.

The Attributes of the User Class is explained below:

- id : This is the unique identification number of the user. It is used as key for the database storage.
- infos: list of info objects that holds the specific information about user for the game.

Info Class

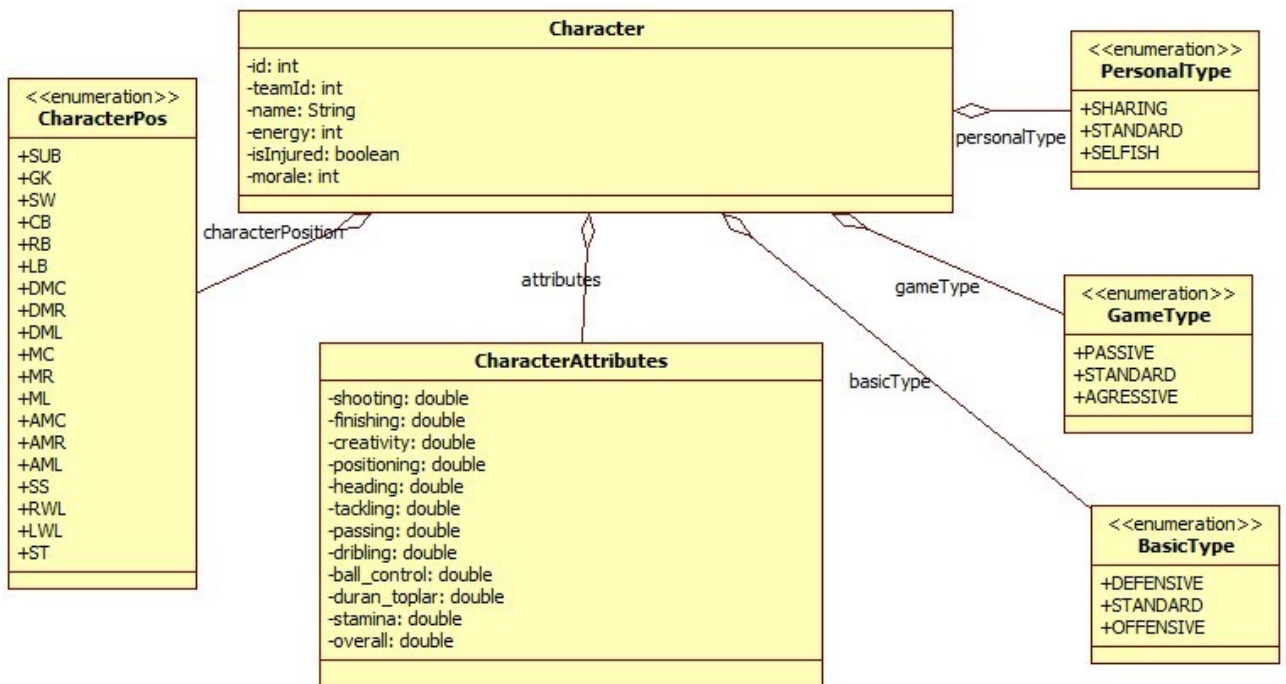
Info class is the abstract parent class for all the information to be defined about the user. The attributes of the info class are:

- id : Holds the unique id of the user.
- name : Holds the name of the information object.

In our product, there are four subclasses of Info class. These classes listed below:

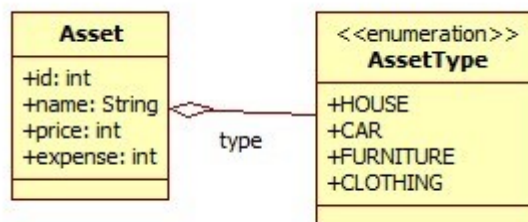
- CharacterInfo: This class holds the information about the character of the user.
- AssetInfo: This class holds information about the user's virtual money and all of its assets of the user.
- UserUniversalInfo: This class holds the information about the user's token and general statistics. The statistic is an abstract class that can be implemented.
- FacebookInfo: This class holds the basic information that can be acquired from the user's Facebook account.

Character (Player) Class



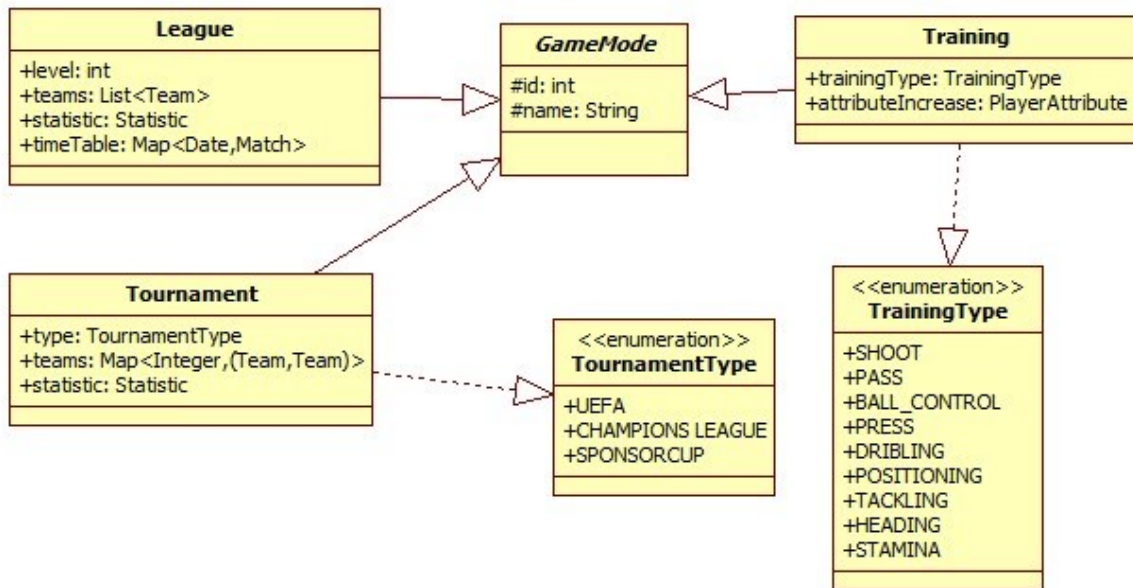
Character class holds the all the information about user’s character which affects the performance of the character inside the game. For example, energy attribute effects how long and how active the character can play in a match.

Asset Class



The Asset class holds the information about an asset that can be bought, traded and auctioned inside the game.

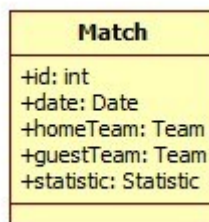
Game Modes



Game mode class is an abstract class for different type of games.

- League class models a standard football league that holds its level, team, statistics and timetable.
- Tournament class models a standard football tournament that holds its type, teams and statistics.
- Training class models a single regular training. It holds the type of training and its attribute increases for the character of the user.

Match Class



Match class models a match between two teams that holds its id, date, teams and statistics.

Team Class

Team class models a football team which includes its id, league id, name, logo id for the predefined list of logos, players, statistics, sponsor id and ids of active tournaments.

Note: Statistic class is an abstract class that is going to be extended and implemented for every class that we have defined above.

4.1.2 Data Dictionary

In the project, we are using a NoSQL database which holds the information in a textual format. Therefore, there is no need for a data dictionary. Storing information to the database will be in the json format of the data objects that we have defined in the previous chapter.

5 Behavioral Model and Description

5.1 Description for Software Behavior

5.1.1 Events

5.1.1.1 Ambit Scene Selected

User will be redirected to his ambit menu (scene).

5.1.1.1.1 Interaction with Player's House

A menu will pop up, which has options to throw a party, rest, make self-training, have recreation time or buy a new house.

5.1.1.1.2 Interaction with Furniture Set

User will be redirected to furniture set shop.

5.1.1.1.3 Interaction with Player's Car

User will be redirected to car shop.

5.1.1.2 Club Life Scene Selected

User will be redirected to club life menu (scene).

5.1.1.2.1 Team Training

A match screen will be opened, where user can follow the match from commentator's speech. The match will be between his teammates. After match ends, match statistics and player's rating will be shown.

5.1.1.2.2 Interaction with Club Administration

List of wishes from club will be appeared. User will be able to choose one of them. The list will include "A bigger stadium", "A better contract" and "Better facilities".

5.1.1.3 Social Life Scene Selected

User will be redirected to social life menu (scene).

5.1.1.3.1 Interaction with Girlfriend

A menu will pop up, which includes possible events with girlfriend. These events are buying a gift to her, dating with her and breaking up with her.

5.1.1.3.2 Interaction with Family/Teammates

User will be able to only go out with his family or teammates. When an interaction occurs with family or teammates' image, a menu will pop up, which includes possible places to go.

5.1.1.3.3 Going Out

A menu will pop up and user will choose, with whom he wants to go out with.

5.1.1.3.4 Charity Work

A menu will pop up and user will choose available charity works with expected outcomes.

5.1.1.4 Manager Room Scene Selected

User will be redirected to manager room menu (scene).

5.1.1.4.1 Search & Inspect Teams

After user goes to search & inspect teams menu, he will see a listing of all teams around the world. He will be able to see every necessary information of teams, such as empty positions of teams, levels of teams and budgets of teams.

5.1.1.4.2 Request Transfer to a Team

The user will be examined by desired club's transfer mechanism and get a result in same day.

5.1.1.5 Character Scene Selected

User will be redirected to character menu (scene).

5.1.1.5.1 Interaction with Player's Clothes/Accessory

User will be redirected to clothes & accessory shop.

5.1.1.5.2 Interaction with Energy Drink

User will be shown a pop up menu, in which he can buy or use an energy drink.

5.1.1.5.3 Interaction with Face style

User will be redirected to hairdresser shop.

5.1.1.6 News Scene Selected

User will be redirected to news menu (scene).

5.1.1.6.1 Interaction with News

If there is any chance to reply the current news, he'll be shown answers as a pop up menu.

5.1.2 States

5.1.2.1 Login Scene

There will be Facebook login button in mobile devices. There won't be any login scene in web client.

5.1.2.2 Main Menu

There will be links to main scenes of the game. These links will be placed in a map view. The buttons are for ambit scene, club life scene, social life scene, character scene, manager room scene and news scene.

5.1.2.3 Ambit Scene

User will see his living area, his house and his car in front of his house. There will be three different background images depending on player's house. These images are suburbs, skyscrapers and mansions.

5.1.2.3.1 Cars Shop

User will see available cars, which are for sale. There is a chance to buy a second hand or a brand new car.

5.1.2.3.2 Furniture Set Shop

User will see available furniture sets, which are for sale. There is a chance to buy a second hand or a brand new one.

5.1.2.3.3 House Shop

User will see available houses, which are for sale. There is a chance to buy a second hand or a brand new one.

5.1.2.4 Club Life

User will see, his team's standings, his team's line up button, his team's stadium image and his team's facilities.

5.1.2.5 Social Life

User will see his girlfriend (if any), his family, his teammates, a button for going out and a button for charity works. He'll also see relation status of all those people. Those status will be a bar image. There will be different images for girlfriend, depending on player's level.

5.1.2.5.1 Girlfriend Relation Status

A pop up menu, which includes these buttons: Buying a gift to her, have a date with her and break up.

5.1.2.5.2 Going out Menu

User will see the list which includes his girlfriend (if there is any), his family or his teammates.

5.1.2.5.2 Charity Work Menu

A pop up menu, in which user can see available charity works to attend.

5.1.2.6 Manager Room

User will see three buttons, which are transfer, finance and sponsorships. He will also see his manager's image, which is also a button that directs him to a shop, where he can change his manager to a new one.

5.1.2.6.1 Transfer Menu

In transfer menu, user will see listing of all teams. He can touch/click a team's name to see a detailed information and request a transfer to related team.

5.1.2.6.2 Sponsorship Menu

User will see his ongoing sponsorship contracts and new offers for sponsorships.

5.1.2.6.3 Investment Menu

User will see his ongoing investments and new investment opportunities. Each ongoing investment will include his daily (real time) profit and each investment opportunity will include risk percentage of related investment.

5.1.2.7 Character Scene

User will see his own character. The character will be able to have different kind of face style, clothes and accessories.

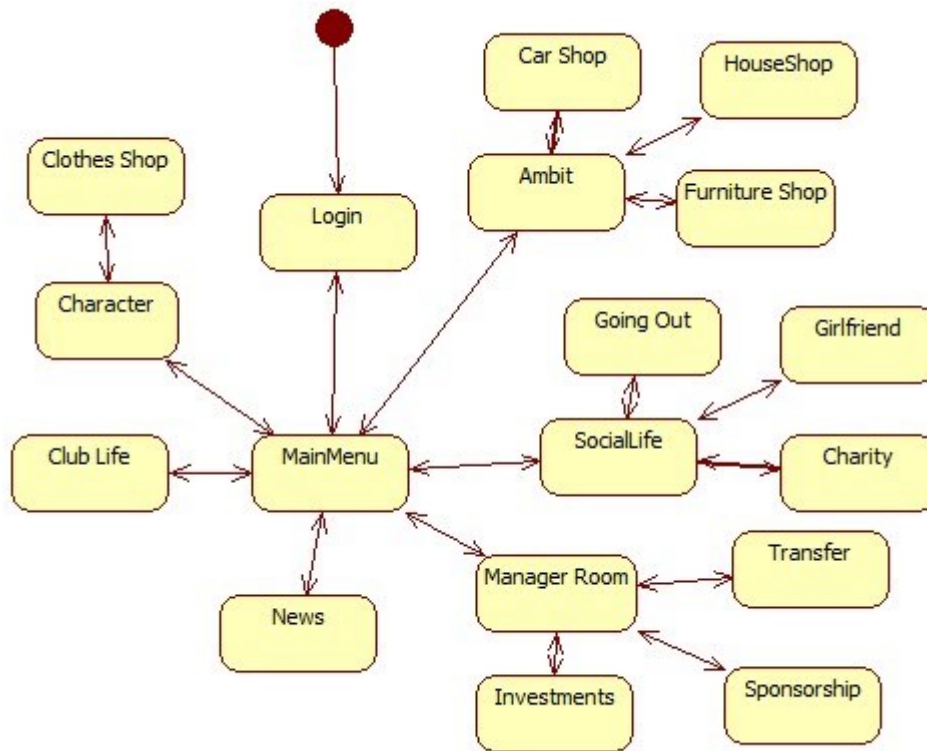
5.1.2.7.1 Clothes/Accessory Shop

User will see available clothes and accessories, which are for sale. There is no chance to buy a second hand.

5.1.2.8 News Scene

User will see a tablet like screen, in which there are news & notifications related to him.

5.2 State Transition Diagrams



6 Planning

6.1 Team Structure

There shall be 5 developers in team. All team members have main focus points but they will work on every stage of development, if it is necessary.

There will be two different stages for system design, which are developing prototype and delivering end product. For prototype phase of development, 4 developers shall work on Java server and 3 developers shall work on web client. The prototype will be available only in web browsers.

For delivering end product phase, 3 developers shall work on Java server, 2 developers shall work on web client, 1 developer shall work on iOS client, 1 developer shall work on Android client and 1 developer shall work on data mining.

6.2 Estimation

This is the weekly basic schedule of the project starting from now.

2. Week– Finishing system design for Java server, starting system design for web client and starting to implementation of Java server.

4. Week– Finishing system design for web client and starting implementation of web client.

8. Week– Finishing both web client and Java server demos and starting to test the system. Also starting to bug fixing in same time with testing.

10. Week– Finishing all current bug fixes.

11. Week– Presentation of the application on web client, taking feedback and starting to add/remove some features. Also, starting to system design of iOS and Android clients.

14. Week– Finishing both iOS and Android clients' system designs and starting to implement these clients. Also starting to improve Java server and web client.

18. Week– Finishing demos of mobile clients and starting to test.

19. Week – Starting to bug fix of mobile clients and starting to improve mobile demos.

23. Week – Finishing Java server and web client, than beginning to test processes of them.

24. Week – Beginning to bug fixes of Java server and web client. Continue to test process of them and fix new bugs.

26. Week – Finishing mobile clients and beginning to test processes of them.

27. Week – Beginning to bug fixes of mobile clients and continue to test process of them and fix new bugs.

29. Week – Delivering the project.

6.3 Process Model

In the first meeting with our sponsors, we were told that we will be following agile software development methodology. In the milestones we will get feedback from customer and continue our process according to their feedback.



7 Conclusion

This software requirements specification document gives information about the related game's interfaces, functionalities, states and events, major data classes, development constraints and planning.

Firstly the game we are creating is shortly described in introduction section. Then the functionalities in the game is mentioned with use cases. Afterwards all the functions the game has is explained one by one in detail. After that non-functional requirements like performance requirements and design constraints are stated. Data models and their associations are shown with a figure. Then behavioral model and description is explained with detailed information of events and states. They are supported with state chart diagram. Finally, we presented our team structure, schedule and process model for this project.