Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

Play and learn

Software Requirements Specifications

<< Version 1.0 >>

Team Names

Ahmed Hesham El-Metwally

Ahmed Mostafa Ismail

Mina Nabil Zaki

Ahmed Gamal Elmoslhy

November& 2016

Contents

[Team 3](#_Toc402452669)

[Document Purpose and Audience 3](#_Toc402452670)

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope 3](#_Toc402452673)

[Definitions, acronyms, and abbreviations 4](#_Toc402452674)

[Requirements 4](#_Toc402452675)

[Functional Requirements 4](#_Toc402452676)

[Non Functional Requirements 6](#_Toc402452677)

[System Models 5](#_Toc402452678)

[Use Case Model 8](#_Toc402452679)

[Use Case Tables 9](#_Toc402452680)

[Ownership Report 17](#_Toc402452681)

# Team:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **E-mail** | **Mobile** |
| 20140048 | Ahmed Hesham El-Metwally | Popup.1996@yahoo.com | 01112982922 |
| 20140045 | Ahmed Mostafa Ismail | Ahmedmostafa.fci16@gmail.com | 01159259192 |
| 20140286 | Mina Nabil Zaki | Minanabil78@yahoo.com | 01228075138 |
| 20140010 | Ahmed Gamal Elmoslhy | ahmed\_midoo250@yahoo.com | 01284233613 |

# Document Purpose and Audience:

Our document presents detailed description about the user interface features functional and non-functional requirement And the use case of the system our audience are analyst, tester, developers and project manager.

# Introduction:

## Software Purpose

The purpose of software is to teach some basic concepts to the students through educational games.

## Software Scope

The scope of the project is to offer some educational games designed and implemented by teachers to ensure on the understanding of the basic concepts and also the gaming experience. We have three categories Run code games, Match pictures games and Multiple choice games.

## Definitions, acronyms, and abbreviations:

Definitions of Actors:

|  |  |
| --- | --- |
| **Actors** | **Definition** |
| **Student** | The student is one of essential Components Of the system. He chooses category of the games, rate it or comment on it and participate in the score board |
| **Teacher** | The teacher is the one who create a game and he can also delete it. He can also access the opinions including the ratings or comments and the score board. |
| **System Game** | It is the kernel of the website where it saves the progress and the scores of the students, games developed by teachers and accounts. |
| **Admin** | He is the one who manages all things in the website and receives complains. |

# Requirements

## Functional Requirements:

The functional recruitment of the system is organized to support two types of users the student and the teacher by the System.

1. **Creating account :**

The student and the teacher should create account verified with E-mail. They can use this account to login to the website every time and the game statistics and progress which is saved to the account.

* **Input**: Username, E-mail, Name, Age, Password and the gender.
* **Processing:** check on the validity of the E-mail and if it is valid it stores the data in the database.
* **Output:** It displays that the account created successfully.

1. **choosing categories :**

The student chooses the category of the game he wish to play either match pictures or multiple choices or run code game, the teacher also choose from these categories to make his game play a game.

* **Input:** the choice either the student or the teacher.
* **Process:** going to this category.
* **Output:** Display the category on that have been chosen either by the student or the teacher to choose a game to play.

1. **play :**

This function allow the user which is either the student or the teacher to play the game after choosing it and it includes:

* Starting the game.
* Stopping the game.
* The action taken through the whole game procedure.
* Exiting the game.

1. **Display score board :**

This function is responsible for showing the score board of the game which contains the top 20 highest scores.

1. **Display profile information :**

This function is responsible for the displaying the information of the profile(name ,games played and its highest score and the progress ) for the student ,And for teacher it display his personal data, the games created if it exist and if he play a game it also will display his progress and scores in the game .

1. **Bookmark game:** If Student play any game, he can bookmark this game. Which makes him play it whenever he wants to play the same game once again.
   * **Input:** It takes the game chosen by the student.
   * **processing :** Saves the game in Student’s book mark
   * **Output:** Add the game to the student’s bookmark list.
2. **See Opinions :**

Display all the Opinions on the selected game+ created by the teacher. The opinions include the rating and the comments.

1. **Add Opinion :**

The student can add opinion to any game where he can add comment and rating to the game.

* **Input :** Takes input from the student either the comment or the rating
* **Processing:** Add the opinion to the database of the game which will appear to the teacher on request.
* **Output:** It displays that the opinion added successfully.

1. **Receive and Solve Complains :**

The admin receives complains of all types from the users and works on solving it and reply to the user again waiting for feedback.

1. **Manage Ethics :**

The admin cares about managing ethics in the website through banning accounts who curse or violates the ownership or Academic honesty.

1. **Approve Game:**

The admin checks for any errors in the game progress or in the implementation of the game. If all is good, the admin approve the game and make it available for the users.

1. **Save Accounts and Games :**

The System saves the data of the accounts and also saves the codes of the games created by the teachers.

1. **Achievements and Scoreboard :**

The System saves the achievements of the student in each game he plays and creates scoreboard for each game containing the top 20.

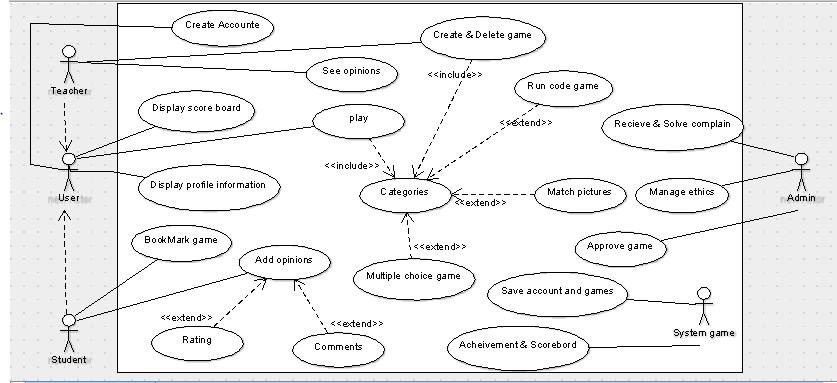
## Non Functional Requirements

It describe the level of requirements that are not related directly to the system and how the system work and we can judge on the system from it as it show the quality of the system as it deal with system architecture.

|  |  |
| --- | --- |
| **Non-Functional Requirement** | **Details** |
| **1.Security** | It manages what everyone should access in data and deal with the privacy of each one logged in the system and protect it from attacks as:   * On logging in it check on the validity of the user name and the password. * The system shall automatically log out all customers after a period of inactivity. * Protects the transferred data through the network. * The data stored should be secured. |
| **2.Performance** | Our website is fast and have high loading speed as it can handle many users using the system without losing any of its speed as most of the response times must be below 3 seconds. |
| **3.Usability** | Our user interface is simple to use for both of teacher and student (of all ages), and shall be compatible with any browsers. |
| **4.Avalability** | Our website is available all the time (24 h) except when maintaining the website. |
| **5.Reliability** | Our website is reliable as it can handle high capacity of users in range of 1000 visitor at the same time without failure or any crash and with good performance. |
| **6.Stability** | Our website is with almost no bugs and least error rate possible. |
| **7.Supportability** | Our admin is responsible for receiving and solving complains send by the user either teachers or students. |
| **8.Modifiable** | Our website can be modified to make any changes in the software. |
| **9.Accessibility** | Any student can play any game freely. |
| **10.Design Tools** | * The web site make some design tools available to teachers. * The teacher are free in using any programming language to make the game. |

## 

## Use Case Model

****

## Use Case Tables:

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Creating Account | |
| Actors: | Student ,Teacher And System | |
| Pre-conditions: | To access the website and create account for the first time. | |
| Post-conditions: | When successfully creating account. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter username, E-mail, Phone number, gender and Password. |  |
|  | 2- System Verify user data |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter username, E-mail, Phone number, gender and Password. |  |
|  | 2- E-mail is not valid or repeated.  3- Password not in the right format+ |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Choosing Category | |
| Actors: | Student and Teacher | |
| Pre-conditions: | Either to play game or create game. | |
| Post-conditions: | When successfully choosing category then it goes to play if it was student or it goes to create game if it’s a teacher. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User choose category. |  |
|  | 2- System will move him to the selected action before. |
| Includes: | 1.Create and Delete Game  2.Play | |
| Extends | 1.Run code Games  2.Multiple choice games  3.Match Pictures | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Play | |
| Actors: | Student and Teacher | |
| Pre-conditions: | To access the function the user must have chosen a category. | |
| Post-conditions: | When the user press the exit button in the game. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User chooses certain game in the category chosen. |  |
|  | 2- System opens the game for the user |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Display Score Board | |
| Actors: | Student and Teacher | |
| Pre-conditions: | To access the Score Board the user must have an existing account. | |
| Post-conditions: | When successfully Displaying the top 20 scores. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User press the button of show score board. |  |
|  | 2- System show the lists of highest 20 scores in the selected games. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Display Profile Information | |
| Actors: | Student and Teacher | |
| Pre-conditions: | To access the function the user must be logged in onto your account. | |
| Post-conditions: | When user press exit and choose other command. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User press Display personal information |  |
|  | 2- System shows name, E-mail, Gender, achievement, mobile and games created if the user was teacher. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Bookmark game | |
| Actors: | Student and Teacher | |
| Pre-conditions: | To access the function the user must have played this game before. | |
| Post-conditions: | When successfully creating account. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User press bookmark this game button |  |
|  | 2- System add this game to the bookmark list of the user in the database of the system. |
| Exceptions: | **User Action** | **System Action** |
| 1- User press bookmark this game button |  |
|  | 2- This game is already bookmarked before. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | See Opinions | |
| Actors: | Teacher | |
| Pre-conditions: | To access the function the user must be teacher and created games before. | |
| Post-conditions: | When successfully showing the opinions and the user press exits. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User press the See Opinions button |  |
|  | 2- System displays the opinions of the other users. |
| Exceptions: | **User Action** | **System Action** |
| 1- User press the See Opinions button. |  |
|  | 2- You haven’t created games yet.  3- There is no opinions yet. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Add Opinions | |
| Actors: | Student | |
| Pre-conditions: | To access the function the user must be student and played the game to add opinion on it. | |
| Post-conditions: | When successfully adding the opinion and exiting this window. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter the comment or the rating. |  |
|  | 2- System stores the data in the database of the opinions list of the selected game. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter the comment or the rating. |  |
|  | 2- You can’t submit the opinion empty, You must write something. |
| Extends | 1.Ratings  2.Comments | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Receive and Solve complains | |
| Actors: | Admin | |
| Pre-conditions: | To access the function the users must have sent complains. | |
| Post-conditions: | When successfully reading the complains by the admin and working on solving it. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Admin press See Complains |  |
|  | 2- System Shows the saved list of complains of all users. |
| Exceptions: | **User Action** | **System Action** |
| 1- Admin press See Complains |  |
|  | 2- There is no complains to show. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Manage Ethics | |
| Actors: | Admin | |
| Pre-conditions: | To access the function the Admin must have received complain about ethical issue. | |
| Post-conditions: | When successfully solving that issue. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Admin press Show Ethical complains button. |  |
|  | 2- System Shows ethical complains. |
|  | 3- Admin takes actions towards users who violated the Ethical rules. |  |
|  |  | 4- System save the actions taken to the database. |
| Exceptions: | **User Action** | **System Action** |
| 1- Admin press Show Ethical complains button. |  |
|  | 2- There is no ethical complains submitted yet. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | Approve Game | |
| Actors: | Admin | |
| Pre-conditions: | To access the function the Admin will check for errors in the progress or the implementation of the game code submitted by a teacher. | |
| Post-conditions: | When the game has no errors then its approved. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Admin checks for errors and if there isn’t any, he approves it. |  |
|  | 2- System starts make the game available for the users. |
| Exceptions: | **User Action** | **System Action** |
| 1- Admin checks for errors and there was errors found then it will not be approved. |  |
|  | 2- The System will inform the teacher that his code wasn’t approved and what was the type of the error. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Save Accounts and games | |
| Actors: | System Game | |
| Pre-conditions: | Other function must have been used such as create account or create game. | |
| Post-conditions: | When successfully saving the data to the data base. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Other function will send the data |  |
|  | 2- System will save the data to the database. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 13 | |
| Use Case Name: | Achievements and Score board | |
| Actors: | System Game | |
| Pre-conditions: | Data is sent from game being played right now and the system saves second by second | |
| Post-conditions: | When successfully saving the data to the data base. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Other function will send the data |  |
|  | 2- System will save the achievements to the database and modify score boards. |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Use case | *Ahmed Mostafa* |
| Nonfunctional requirements | *Ahmed Gamal El Moslhy* |
| Functional requirements | *Mina Nabil, Ahmed Gamal* |
| Use case tables | *Ahmed Hesham , Ahmed Mostafa* |
| Introduction, Scope and Definitions | *Ahmed Hesham , Mina Nabil* |