Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

**Funny learning ☺**

Software Requirements Specifications

Hanaa Salah

Marwa Ashraf

Renad Salem

Zeinab Ibrahim

November 2016

Contents

[Team 2](#_Toc466135031)

[Document Purpose and Audience 3](#_Toc466135036)

[Introduction 3](#_Toc466135037)

[Software Purpose 3](#_Toc466135038)

[Software Scope 3](#_Toc466135039)

[Requirements 4](#_Toc466135040)

[Functional Requirements 4](#_Toc466135041)

[System Models 5](#_Toc466135042)

[Use Case Model 5](#_Toc466135043)

[Use Case Tables 5](#_Toc466135044)

[Ownership Report 17](#_Toc466135045)

# 

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140260 | Marwa Ashraf | Merotreka212@gmail.com | 01140095542 |
| 20140132 | Zainab Ibrahim | Zeezee3215@gmail.com | 01025898706 |
| 20140129 | Renad Salem | renadsalem97shr@gmail.com | 01000167891 |
| 20140310 | Hanaa Salah | Hanasalah1512@gmail.com | 01122909949 |

# Document Purpose and Audience

-This document is about our software project , that include all details for the software project.

-Document contain requirements (functional ,non- functional ), use case model and use case tables.

-This is like a contract between us and the client (T.A ) , we can go back to it when we argue about something or any detail.

Our audience:

-Client.

-Developers and designers.

-Project Managers

# Introduction

## Software Purpose

-A service to enhance the creativity of teachers where they can create a new way to teach the student, and the creativity of student in solving. We also apply the active learning (on hands training).On that way we develop the educational operation which makes a greater output.

## Software Scope

-Our project is a website and the clients have to create an account on this website.

-Our game categories (Matching pictures - multiple choices – complete).

-Our games fields are mathematical, programming, science, historical and languages.

# Requirements

## Functional Requirements

Our project allows the users to:

1-Create accounts.

2- Log in.

3-Creating games.

4- Editing games.

5- Removing games.

6- Playing games.

7-Comment.

8- Replying to comments.

9- Rating games.

10-Choose field and category for playing.

1-Create account

Student and teacher must create account to use the website and the system will verify confirmation to secure his/her data.

2- Log in

Student and teacher can log in by enter his/her email and password, the system get data and verify it

3-Creating games

Only teachers have a permission to create new games, he /she must explain how to play it and choose it's field and category then upload the game.

4- Editing games

Only the teacher how create the game can edit it , he /she must select the game to edit it and the system will confirm the ownership of the game then the teacher can upload the edited game on it's category and field .

5- Removing games

Only the teacher how create the game can remove it, he/she must select the game and the system will confirm the ownership of the game then the teacher can delete it

6- Playing games

Teacher and student can play any game on the website, he/she must choose the field and the category for the game then can start to play it, the system will list its levels and the top 20 for the game, the system will compute the score and store it after every level on the game.

7- Comments

Teacher and student after playing the game can make comment for the owner of the game ,the system will add the comment to the other comments and send notify for the owner to view it , User who didn't play the game can't able to make comment .

8- Replying to comments

Teacher and student can reply to any comment for other users, User how didn't play the game can't reply for comments.

9- Rating games

Teacher and student after playing the game and finish some levels on the game ,the system will ask the user to rate the game then the user can rate and write his/her opinion or not ,the system will notify the owner of the game .

10-Choose Game

Teacher and student before playing any game must choose the field, the system will list the categories then he/she can choose the category then play the chosen game.

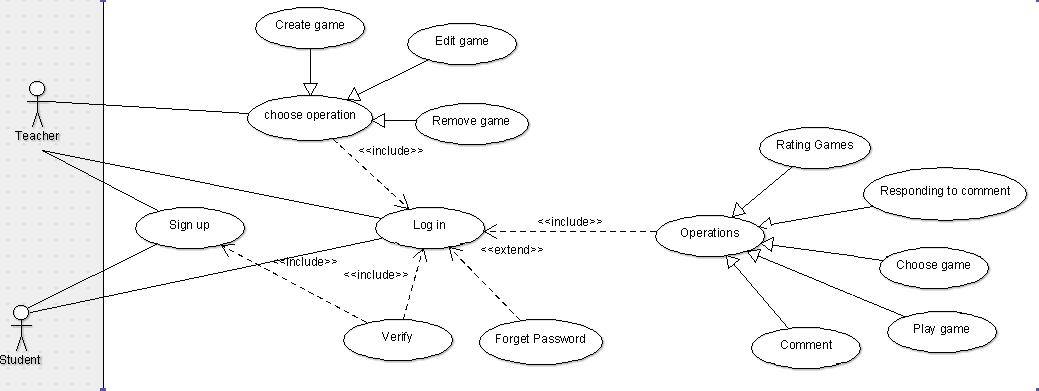
**Non Functional Requirements**

|  |  |
| --- | --- |
| **Security** | We secure information of website by policies to prevent the unauthorized access or modification, and secure users data and we apply that by the simple way of allocating each user unique name and password. |
| **Accessibility** | The website can be accessed for all users by only writing the URL |
| **Usability** | It's easy to be used by anyone as it's clear, not complicated as it’s way in login or sign up is as usual and it’s categorized into categories so any one can understand it and deal with it easily |
| **Reliability** | The website won't be down unless there is a special case and in such case it won’t be down more than 24 hours |
| **Performance** | The website has little response time as it takes about 10 seconds to log the user in and a page render time is about 15 seconds |
| **Maintaining** | Handling the problems of the website without any damage in the user's data |
| **Scalability** | The web can be used by 1000+ users at the same time ,the website is flexible to have other properties and fields |
| **Privacy** | We apply privacy be giving each user the choice of making any of his personal information private or public |

# 

# System Models

## Use Case Model



## 

## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Create Account | |
| Actors: | Teacher-Student | |
| Pre-conditions: | Teacher or Student not has Account on the website | |
| Post-conditions: | The Account created and user can use it | |
| Flow of events: | **User Action** | **System Action** |
| 1- user click to create Account . |  |
|  | 2- System Ask the user if he/she student or teacher. |
| 3- User fill the information such as name ,Age ,E-mail and Password and click to button Create. |  |
|  | 4- System will send confirmation to the email to confirm the Account . |
| 5- User is ready to play on the website. |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter wrong Email or email used before in the website. |  |
|  | 2- System will send error message contain this Error  3- System can't accept this account |
| Includes: | The Account will be created and opened to the user to start the games and choose the categories | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Log in | |
| Actors: | Teacher  Student | |
| Pre-conditions: | The user has an existed account. | |
| Post-conditions: | Verification then opening the user's account. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enters the e-mail and password. |  |
|  | 2- System verifies the account.  3-system allows the user to open the account. |
| Exceptions: | **User Action** | **System Action** |
| 1- User forget his password. |  |
|  | 2-system goes to the user's e-mail and sends verification code. |
| Includes: | Existed account. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Choose game | |
| Actors: | Teachers  Students | |
| Pre-conditions: | User login to the web. | |
| Post-conditions: | User choose the game to play it. | |
| Flow of events: | **User Action** | **System Action** |
| 1- user chooses a field |  |
|  | 2- system list categories in the field |
| 3- user chooses the category |  |
|  | 4- System lists all games in the category. |
| 5-User chooses a game. |  |
|  | 6-System starts the game for the user to play. |
| Exceptions: | **User Action** | **System Action** |
| 1-log in. |  |
|  | 2- user forget his password. |
| Includes: | Log in. | |
| Notes and Issues: | -------- | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Play game | |
| Actors: | Teachers – Students | |
| Pre-conditions: | User login and choose the game | |
| Post-conditions: | System computes the score of the player and update the list of top 20 | |
| Flow of events: | **User Action** | **System Action** |
| 1- User choose the field. |  |
|  | 2- system list categories in the field. |
| 3- User choose the category. |  |
|  | 4- System list all games in the category. |
| 5-User choose the game , user can play or list the top 20 in this game, user start from the last level he had finished else he will start from the first level and so on |  |
|  | 6-System will list top20 in the game and save each completed level with its score |
| Exceptions: | **User Action** | **System Action** |
| 1- User Search for unavailable game |  |
|  | 2- Game is not found. |
| Includes: | Login . | |
| Notes and Issues: | Top 20 will not include teachers . | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Comment | |
| Actors: | Teachers  Students | |
| Pre-conditions: | 1- the user have existed account  2- The user has played the game to be able to comment for it. | |
| Post-conditions: | -Updating the comments by adding the user’s comment. | |
| Flow of events: | **User Action** | **System Action** |
| 1- After playing the game the user write his comment. |  |
|  | 3-System adds the user's comment to other comments.  4-System sends a notification to the owner of the game. |
| Exceptions: | **User Action** | **System Action** |
| 1-User didn’t play the game. |  |
|  | 2- unable to comment. |
| Includes: | play the game. | |
| Notes and Issues: | ------- | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Responding to comments | |
| Actors: | Teachers  Students | |
| Pre-conditions: | Owner of the games receives a notification. | |
| Post-conditions: | -Adding teacher reply | |
| Flow of events: | **User Action** | **System Action** |
| 1-teacher receives a notification. |  |
| 2-teacher can reply to the comment. |  |
|  | 3-System adds the reply to the comment.  4-System sends a notification to the student who wrote the comment. |
| Exceptions: | **User Action** | **System Action** |
| 1-if another teacher or user reply. |  |
|  | 2- unable to reply. |
| Includes: | Log in and receiving a notification. | |
| Notes and Issues: | ------ | |

|  |  |
| --- | --- |
| Use Case ID: | 7 |
| Use Case Name: | Create Game |
| Actors: | Teacher |

|  |  |  |
| --- | --- | --- |
| Pre-conditions: | Teacher already has an Account on the website and login from it. | |
| Post-conditions: | Game will be added on the correct category. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User clicks to Create Game and choose the field and the category for this game. |  |
|  | 2- System open page to the user to explain his/her game and Write about How to play it for students. |
| 3- User uploads the game. |  |
|  | 4- System set the game on the owned category and the specified field and user is ready to play. |
|  |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User upload game has bugs or error (there is a problem on the format of the game). |  |
|  | 2- System can't accept the game and send error message. |
| Includes: | Log in and verification is a teacher. | |
| Notes and Issues: | Not allowed for student to edit or upload any game | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Edit game | |
| Actors: | Teacher | |
| Pre-conditions: | -Game’s existence  -Ownership of game | |
| Post-conditions: | Game is updated correct | |
| Flow of events: | **User Action** | **System Action** |
| 1-Teacher choose edit game |  |
|  | 2-System confirms Ownership. |
| 3-Teacher select a field |  |
|  | 4- System list all categories |
| 5-Teacher select category |  |
|  | 6- System list all games |
| 7- Teacher select game |  |
| 8-Teacher edit game and upload the edited game and write the new updates . |  |
|  | 9- System updates the game |
| Exceptions: | **User Action** | **System Action** |
| 1- if another teacher or student edit game |  |
|  | 2- User is not authorized. |
| Includes: | Log in . | |
| Notes and Issues: | \_\_\_\_\_\_\_\_\_\_\_\_\_ | |
| Use Case ID: | 9 | |
| Use Case Name: | Remove game | |
| Actors: | Teachers | |
| Pre-conditions: | -Game’s existence  -Ownership of game | |
| Post-conditions: | Verification message with validation of editing that game | |
| Flow of events: | **User Action** | **System Action** |
| 1- User select remove game |  |
|  | 2- System confirms Ownership. |
| 3-User select field. |  |
|  | 3- System list all categories |
| 4- User Select category |  |
|  | 5- System list all games |
| 6- User select the desired game to remove. |  |
| 7- User remove game |  |
|  | 8- System verify the game’s deletion |
| Exceptions: | **User Action** | **System Action** |
| 1- if another teacher or student delete a game. |  |
|  | 2- User is not authorized. |
| Includes: | Log in. | |
| Notes and Issues: | \_\_\_\_\_\_\_\_\_ | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Rate | |
| Actors: | Students  Teachers | |
| Pre-conditions: | Student and teachers first must play the game | |
| Post-conditions: | System asks the player to rate the game. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User chooses the game and plays it and finishes the level he was playing. |  |
|  | 2- System asks the player to rate. |
| 3- User can rate the game and write his opinion about the game. |  |
|  | 4- System sends a notification to teacher that created the game that someone rated his game. |
| Exceptions: | **User Action** | **System Action** |
| 1- If user doesn’t play so he can’t rate any game. |  |
|  | 2-System doesn’t show any thing. |
| Includes: | Log in and play the game to rate it. | |
| Notes and Issues: | Write your opinion according to our rules for respecting the developers. | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Document purpose ,software scope and non-functional  Use case model | *Hanaa salah*  *Renad salem* |
| Use case tables  Requirements | *Zeinab Ibrahim*  *Marwa Ashraf* |