Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

**LEARNO**

Software Requirements Specifications

**Aya Mohammed Sayed Gad**

**Sara Mahmoud Mohamed**

**Radwa Ali Ahmed**

**Amany Ashraf El Daly**

**Aya Tarek Abd El Nabi**

November 2016

Contents

[Team 3](#_Toc402452669)

[Document Purpose and Audience 3](#_Toc402452670)

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope 4](#_Toc402452673)

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

[Requirements 4](#_Toc402452675)

[Functional Requirements 4](#_Toc402452676)

[Non Functional Requirements 4](#_Toc402452677)

[System Models 5](#_Toc402452678)

[Use Case Model 5](#_Toc402452679)

[Use Case Tables 5](#_Toc402452680)

[Ownership Report 6](#_Toc402452681)

[Policy Regarding Plagiarism: 6](#_Toc402452682)

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140088 | Aya Mohammed Sayed Gad | ayoyagad96@gmail.com | 01140132399 |
| 20140121 | Radwa Ali Ahmed | radwafci2014@gmail.com | 01006224168 |
| 20140068 | Amany Ashraf Abdeltawab | m6me666@yahoo.com | 01142773139 |
| 20140139 | Sara Mahmoud Mohamed | Saramahmoud251@gmail.com | 01276010818 |
| 20100085 | Aya Tarek Abd El Nabi | A1996tarek@gmail.com | 01159683637 |

# Document Purpose and Audience

* **This document list all functional and nonfunctional requirements that the customer need to know, the stake holders that are affected by the system and explain the use cases and their flow of events .This document describe the software purpose and scope of the system in details.**
* **This document expected to be read by customers, project managers, the analytics, the designers and the developers.**

# Introduction

## Software Purpose

The purpose of our software is to help students with understanding the basic concepts of different subjects and to be able to interact with their teachers through educational games.

## Software Scope

* This system is a web application and it’s major features includes :
* Playing Games by students.
* The student has ability to rate games and leave a comment.
* Teachers can create games and test them.
* Teachers can reply to student’s comments.
* Teacher has the ability to edit or remove his/her own games.
* Games are categorized into MCQ , Run Codes and matching Pictures .

## Definitions, acronyms, and abbreviations

|  |  |
| --- | --- |
| **Term** | **Definitions** |
| **MCQ** | **Multi Choice Questions** |
| **Run Codes** | **Coding games to help students understand basics of programming concepts.** |
| **Identification Code** | **A Code that verify the identity of the teacher and make sure he has the privileges to create, remove or edit games.** |
| **Monitor** | **The job of watching , noticing or controlling particular things.** |

# Requirements

## Functional Requirements

**Sign Up into the website:**

* (User must enter a username and a password )
* Students are required to enter some basic information that include their name, age, gender and level of education then submit it .
* System automatically initializes the student’s score to zero.
* Teachers are required to enter some basic information that include their name , age ,gender , Identification Code ,and his/her Educational specialization then submit it.

**Teacher Creates game:**

* Choose specific category to perform operations on.
* Follow the category criteria’s of creating a game.
* Test the game and Launch it.

**Students play Games:**

* Choose specific category of a game to play.
* Start playing a game.
* The Student’s score is updated automatically after he/she finishes the game.
* A rating bar appear to the student to rate the game (optional).
* Student can leave a comment.

**Teachers Monitor Games:**

* To remove a game, the teacher chooses the specific game and click on the delete button which will delete the game and all the comments on it.
* To edit a game, the teacher chooses the specific game and click on the edit button which will allow the teacher to edit the game then launch it again.
* Teachers can reply to the student’s comment.
* Teachers can view the student’s score and progress.

**Games:**

* Selected Game starts while connecting to the Player account.
* Number of questions appears to the player and he /she enter the answer, if answer is true the game score is updated and if it’s wrong, the correct answer appears to the user and a new question is generated.
* The game level updates automatically according to the player’s level and the level of questions increases.
* The comments on the games are shown in order from newest to oldest.

## Non Functional Requirements

**Usability:**

* This Web Application is targeting a wide user community and it comes in a very easy way to use and interesting way of interaction between the system and its users.

**Reliability:**

* This Web Application is available 24/7.
* The probability of error occurrence is low which raises the user’s satisfaction.

**Performance:**

* Sign up operation should take up to 7 minutes for any user.
* Game’s loading time ranges from (40 to 60) seconds.
* Average time of creating a game is 30 minutes.
* Error messages with a brief description and steps of solutions appear to users in a matter of 10 seconds after error occurrence.
* Notifications are sent to the teacher simultaneous when students comments on a game.

**Security:**

* Users must enter a valid unique username and a valid password which the system checks implicitly according to our security Criteria.
* Users must verify their accounts by an active Email then unique code is sent to the email to make sure of the identity of user.
* The Assurance of the identity of the teachers is a must in our system so for each teacher required to have a unique identification code which generated by our developers after reviewing Teacher’s degree and Position.
* Back Up process is performed weekly to make sure that the user’s data are safely stored.

**Scalability:**

* System is expected to support up to 1000 multiple users.
* System is expected to support up to 500 simultaneous players at same game.
* System is expected to support creation up to 15 new games per day.
* System is expected to support simultaneous parallel Comments.

**Portability:**

* System is a web application that can run on different types of platforms and is implemented in java and Some PHP pages.

**Maintainability:**

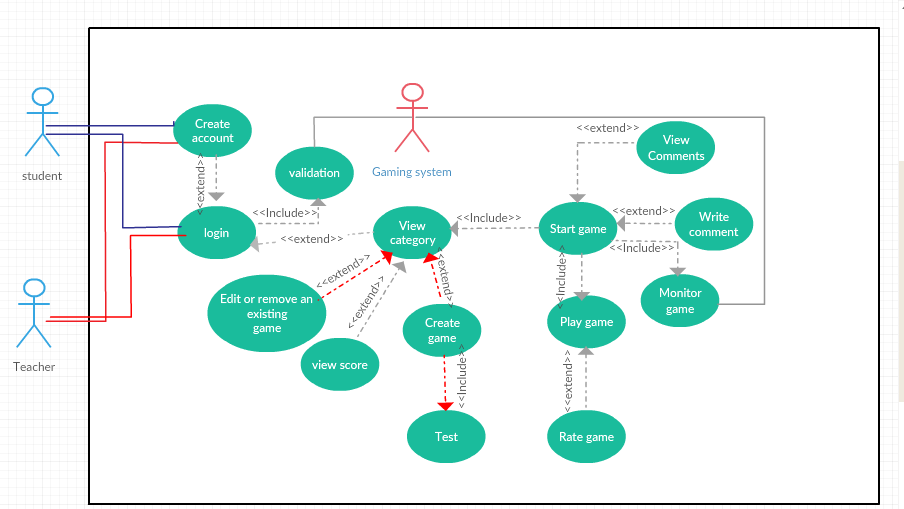
* User’s Rate is used as a feedback to report bugs and errors to take farther actions.
* System goes through maintains cycle every 3 months to update and fix any issues.
* Breakdowns are handled by a specific team of developers.
* Updates make sure to make the system more reliable, efficient and safe.

**Privacy:**

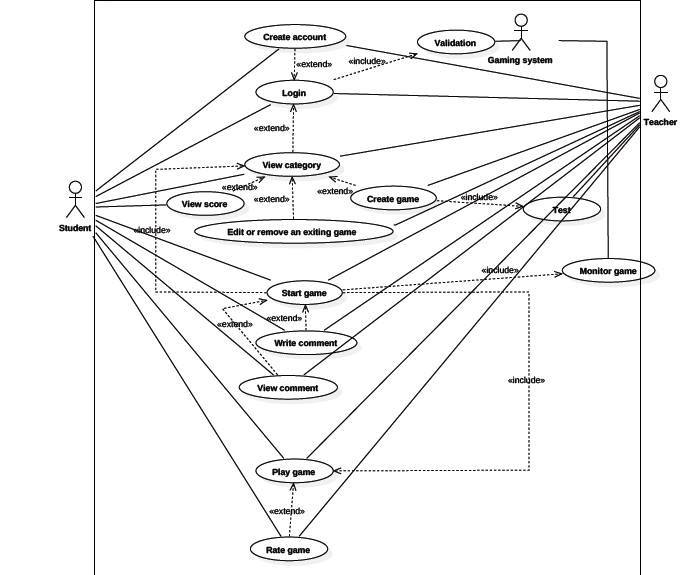
* User’s private Information is secured and can’t be viewed by others unless the user allows it.
* User’s score is only shown by the teacher as a default but the user can edit it to be shown by other users or keep It private.

# System Models

## Use Case Model



* The same use case model



## Use Case Tables:

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Create Account | |
| Actors: | Teacher , Student | |
| Pre-conditions: | Connect to the website link | |
| Post-conditions: | User Account created so Teacher and Student can login | |
| Flow of events: | **User Action** | **System Action** |
| 1- User Enter some basics information. |  |
|  | 2- System Verify user data  3- System store user data |
| 4- User login to the website |  |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter some basics information. |  |
|  | 2- Data is invalid.  3- System rejects Data. |
| Includes: | \_ | |
| Notes and Issues: | System validates user information and gives the authority to create games only for teachers. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | View Categories | |
| Actors: | Students – Teacher | |
| Pre-conditions: | User must login to the system and choose one of the several existing categories. | |
| Post-conditions: | The user can play / create a game in specific category. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enters user name and password. |  |
|  | 2- System Verify user data and display for user specific category of a game. |
| 3-User choose specific category from several categories such as: Run code, Match pictures, MCQ to play or to perform operations like: Creating the game. |  |
| **User Action** | **System Action** |
| Exceptions: | 1-User select category. |  |
|  |  | 2- Network connection Problem.  3- System rejects to display the categories.  4- System request to check network connection. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Edit or Remove a game | |
| Actors: | Teacher | |
| Pre-conditions: | This game actually is created by this teacher. | |
| Post-conditions: | New version of this game is launched / the game disappears from the website. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Teacher choose the category which a game in it. |  |
|  | 2- Display the correct category. |
| 3-Teacher enters to the game and select edit or remove button. |  |
|  | 4-System ensures that this user is a teacher. |
| 5- Teacher makes the operation which he wants to do. |  |
|  | 6-If the teacher edit in a game, the system save the change but if the teacher removes a game the system delete all information that’re related to this game. |
| Exceptions: | **User Action** | **System Action** |
| 1-Teacher edit or remove the game. |  |
|  | 1. The system can't implement the operation. 2. System check connection. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | View student score | |
| Actors: | Teacher-Student | |
| Pre-conditions: | Student must play any game at least one time. | |
| Post-conditions: | Student can view his score and the teacher can evaluate the student by checking his score. | |
| Flow of events: | **User Action** | **System Action** |
| 1-Actor choose the game that the score on it. |  |
|  | 2- The system Access the request and enable the user to enter this game. |
| 3- Actor can view the score. |  |
|  | 4-The system Display the score. |
|  |  |
|  |  |
| Exceptions: | **User Action** | **System Action** |
| 1- Actor trying to view a score related to a game which the student never plays it any more. |  |
|  | 2- System request to check network connection. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Create Game | |
| Actors: | Teacher | |
| Pre-conditions: | User log in as a teacher. | |
| Post-conditions: | User can play this game and the teacher can test the game. | |
| Flow of events: | **User Action** | **System Action** |
| 1-Teacher starts to create game. |  |
|  | 2- Display for teacher all categories. |
| 3- Teacher choose specific category to create a game. |  |
|  | 4-Display the category criteria of creating a game to be followed by teacher. |
| 5- Follow the category criteria of creating a game. |  |
| 6-Test the game and Launch it. |  |
|  |  | 7- Save all information about this game. |
| Exceptions: | **User Action** | **System Action** |
| 1-Teacher creates a game. |  |
|  | 2- Network connection Problem.  3- System rejects saving the game.  4- System request to check network |
| Includes: | Test game | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Test a game | |
| Actors: | teacher | |
| Pre-conditions: | The game is already created | |
| Post-conditions: | The game can be launched | |
| Flow of events: | **User Action** | **System Action** |
| 1-Teacher chooses the created game to test it. |  |
|  | 2-System opens the game. |
| 3-Teacher plays the game and record any Bugs or Failure then fix it. |  |
|  | 4-System Save teacher's edit and launch the new version. |
|  |  |
| Exceptions: | **User Action** | **System Action** |
| 1- Teacher tests the game. |  |
|  | 2-System reloads the game.  3- System request to check network connection. |
| Includes: | \_ | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Start Game | |
| Actors: | Student Or Teacher | |
| Pre-conditions: | Select a specific category. | |
| Post-conditions: | Actor plays a game then his/her score is updated. | |
| Flow of events: | **User Action** | **System Action** |
| 1-Click the start button |  |
|  | 2- System load the Game |
| 3- Start Answering the Questions |  |
|  | 4- System check Answers |
| 5-Finish Game |  |
|  |  | 6-System Update Score and save it. |
| Exceptions: | **User Action** | **System Action** |
| 1- Actor Start Game |  |
|  | 2- Game Crash  3- System shows Error Message with detailed instructions to solve it.  4-System try reloading the game. |
| Includes: | System Monitoring The Game \_ user selects specific category. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Play Game | |
| Actors: | Student or Teacher | |
| Pre-conditions: | Start a game from a specific category. | |
| Post-conditions: | Rate game or leave a comment. | |
| Flow of events: | **User Action** | **System Action** |
| 1-Actor chooses a Subject |  |
|  | 2- System shows a Question. |
| 3- Actor Answer the Question. |  |
|  | 4- System Verify Answer |
|  | 5-Actor Move to Next Question |  |
|  |  | 6-System show next Questions |
|  | 7-Actor Finish Playing |  |
|  |  | 8-System Save score and exit game. |
| Exceptions: | **User Action** | **System Action** |
| 1- Actor Play Game |  |
|  | 2- Network connection Problem.  3- Score is not saved.  4-System request to check network connection.  5-System reloads the game. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Rate a Game | |
| Actors: | Student | |
| Pre-conditions: | Playing a specific game. | |
| Post-conditions: | Choose Other game to play or exist the application | |
| Flow of events: | **User Action** | **System Action** |
| 1-Student click on Rating Bar |  |
|  | 2- System ready to count number of stars given. |
| 3- Student choose number of stars which represents rating. |  |
|  | 4- System Save Student’s rate. |
| Exceptions: | **User Action** | **System Action** |
| 1-Student click on Rating Bar. |  |
|  |  | 2-Rating Bar is frozen.  3-System Reload Page. |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | View Comments | |
| Actors: | Student or Teacher | |
| Pre-conditions: | Start a game. | |
| Post-conditions: | Waiting for a new Reply. | |
| Flow of events: | **User Action** | **System Action** |
| 1-User Choose Game |  |
|  | 2- System Show notification for any one replied to the user previous comments. |
| 3- User Read Comment and reply to. |  |
|  | 4- System Save new reply and adds it to the list of comments. |
| Exceptions: | **User Action** | **System Action** |
| 1-User Show Comments |  |
|  |  | 2- Network connection Problem.  3- Comment not appearing.  4-System request to check network connection. |
| Notes and Issues: | Showing Comments include the owner’s name, time and date of its submission. | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | Write Comment | |
| Actors: | Student | |
| Pre-conditions: | Start a game. | |
| Post-conditions: | Comment submitted. | |
| Flow of events: | **User Action** | **System Action** |
| 1-Student goes to comments section |  |
|  | 2- Text Area appears to user to leave a comment with his/her name included. |
| 3- Student leaves a comment then submits. |  |
|  | 4- System adds comment at the list with the other comments including owner, date and time of its submission. |
| Exceptions: | **User Action** | **System Action** |
| 1- Student add comment |  |
|  | 2- Network connection Problem.  3- Comment not saved.  4-System request to check network connection. |

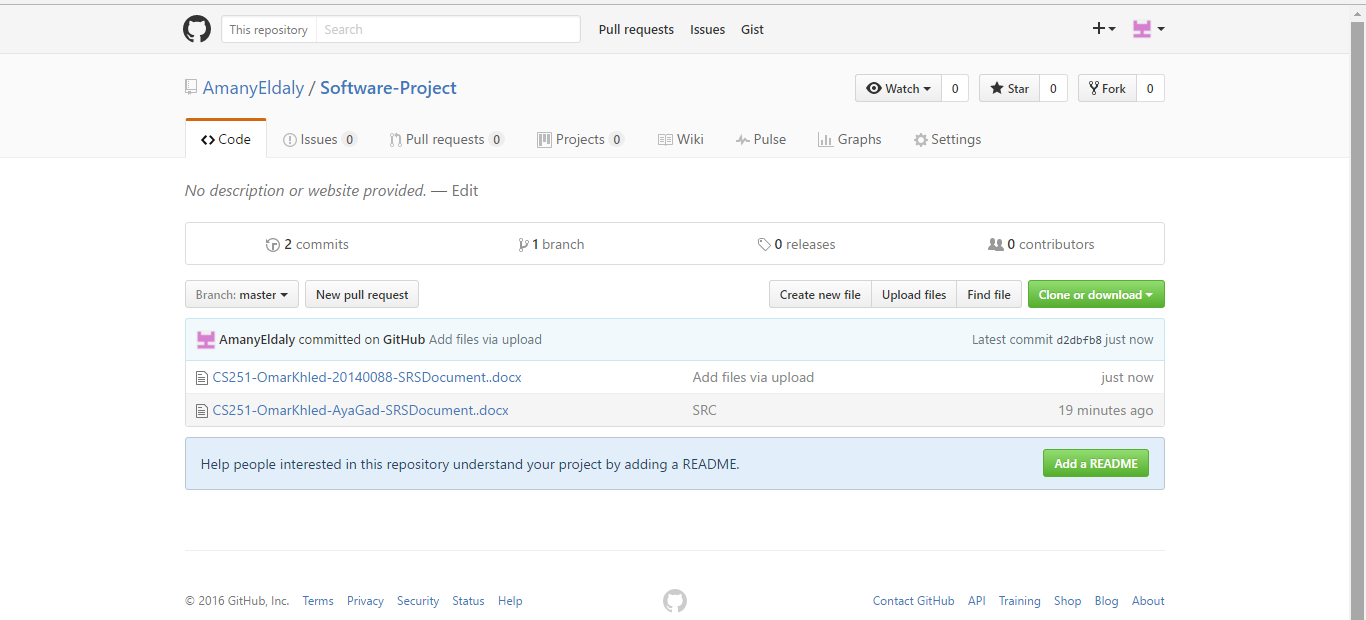
|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Monitor Game | |
| Actors: | System | |
| Pre-conditions: | User starts playing game. | |
| Post-conditions: | System update user score | |
| Flow of events: | **User Action** | **System Action** |
| 1- User starts a game. |  |
|  | 2- System creates a list of Questions.  3- System shows a random Question to the player. |
| 4- User Select /Enter correct answer. |  |
|  |  | 5- System validates the answer.  6- System updates the user score.  7- And so on. |
| Exceptions: | **User Action** | **System Action** |
| 1- User Select /Enter wrong answer |  |
|  | 2- System rejects the answer and shows the correct answer to the user. |

# Complexity Rank:

|  |  |
| --- | --- |
| Create account | 3 |
| Select Category | 1 |
| Edit or remove game | 3 |
| View score | 1 |
| Create game | 4 |
| Play game | 3 |
| Test Game | 3 |
| Start game | 2 |
| Rate game | 2 |
| Monitor game | 5 |
| Write comments | 3 |
| View comments | 2 |

# Ownership Report:

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Document Purpose and Audience. | All members. |
| Software Purpose , Software Scope, Definitions, acronyms, and abbreviations | All members. |
| Functional Requirements. | All members. |
| Non-Functional Requirements | All members. |
| Use Case Model. | Amany, Aya. |
| Use Case Tables. | All members :  Sara (Select Category, Create game, Test, Edit or Remove, View score).  Aya (Login, Create account, Validation, Monitor).  Radwa(Start game, Write comment, Play game, Rate game, view replayed comment) |



Link--🡪 https://github.com/AmanyEldaly/Software-Project