Cairo University  
Faculty of Computers and Information



**Software Engineering I**

**Phase 1**

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| ***ID*** | ***Name*** | ***Email*** | ***Mobile*** |
| 20140001 | Abraam Emad Reda | abraamaj27@gmail.com | 01278055444 |
| 20140004 | Ahmed Ibrahim Mokhtar | ahmed0123031@gmail.com | 01225068308 |
| 20140177 | Ali Mohamed EL-Amin | Ali.amin823@gmail.com | 01112794151 |

Contents

[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)

# 

# Introduction

## Software Purpose

Web Application to help Student to understand basic educational concepts by playing simple games.

## Software Scope

This website is created to help student to understand basic educational concepts by playing simple games   
the student can enter to the website to sign up an account and sign in.

student choose category of the games then choose the game and he can comment and rate about this game

teacher can sign up an account and sign in .

he can create , try , edit and remove any game and reply on student’s comments

## Definitions, acronyms, and abbreviations

Student : sign up an account to understand any educational concept by playing games

Teacher : can sign up to create games to simplify any education concept for students.

Games : methods is created to deliver educational concepts to students and it’s enjoyable.

# Requirements

## Functional Requirements

* ***Sign up***

Student/Teacher enter to the website , then fill his information , to create his account

* **Log in**

Student/teacher enter to site then write his username and password and the system check that’s valid or not .if valid the student/teacher is entered to his account

* **Play game**

Student choose category of the games then system display the games of this category then student choose the game .after student finish the game system display the score of student and save it in the student account .

* **Rating**

1Student click to rate the game then system display the options of rating **(interesting, normal, boring)** then student choose the rating then system display the rating and save it .

* **Comments**

Student click to comment on the game then system display a place of comment then student write his comment and click enter, system save it .

* **Log Out**

Student/Teacher click log out then the system save all changing and log out from the Account .

* **Create game**

Teacher click to create a game. system displays the categories of games then teacher choose the suitable category “Match pictures” , “Multiple Choices game” , “Run code game” and add the game

* **Try game**

Teacher click to try to test the game .the system displays the games which created by teacher then teacher choose the game he want to try if its run correct or not .if the game not valid click to edit it

If correct click finish and the system log out from the game

* **Edit game**

Teacher click to edit to fix the game .the system displays the games which created by teacher then teacher choose the game he want to edit ,then teacher click finish and the system save it and log out from the game

* **Remove game**

Teacher click to remove the game .the system displays the games which created by teacher then teacher choose the game he want to remove .then teacher click finish and system save it .

* **Respond to student comments**

Teacher click to view comments the game , the system displays the games which created by teacher then teacher choose the game he want to view student’s comments , if teacher want to reply click to reply to the comment and the system display a place to comment , teacher write his comment and click enter , the system save it .

## 

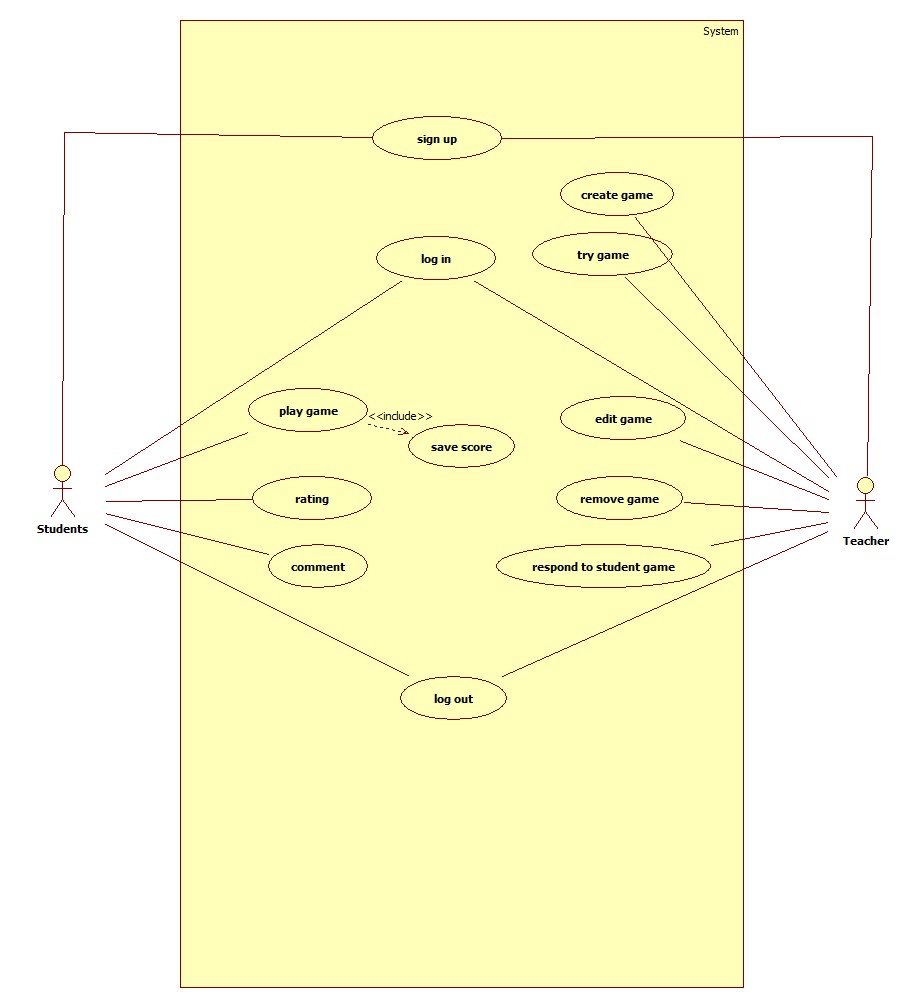
## Non Functional Requirements

|  |  |
| --- | --- |
|  | **Details** |
| **Performance** | * **Short response time**. The loading time of the Web application must be smaller than 10 seconds. All other response times (the game loading) must be below 5 seconds. |
| **Scalability** | * System should be able to support up to 1000 students on one game |
| **Usability** | * **Simple to play**. The game rules of games must be intuitive. A student should learn how to play games to understand it and understand the educational concepts . |
| **Reliability** | * The site must be available any time for accessing |
| **Security** | * Student can play a game if he registered in the website and have confirmation from the teacher which created this game. |
| **Maintainability** | * Updating the site every 3 months if there is any problems fixed it or update the view of the sites. |
| **Implementation** | * **Programming language**. The Website must be implemented in PHP and MySql . |
| **Interface** | * **Simple user interface**. The user interface of website and the games should be understandable to the student. The user interface is based on a main window, which includes a playing field, The button bar offers control buttons for starting, stopping ,rating and commenting on a game for student.and The button bar offers control buttons for creating ,removing ,editing ,trying and view student’s comments . |
| **Legal** | * **Cheap Game**. The development of the websites must be done with a budget 1000$ . As the game should reach as many students and teachers as possible and it should be financed over advertisement, it cannot be expensive. |

# 

# System Models

## Use Case Model



## *Use Case Tables*

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Sign up | |
| Actors: | Student , Teacher | |
| Pre-conditions: | Enter to website | |
| Post-conditions: | Enter to his account | |
| Flow of events: | **User Action** | **System Action** |
| 1. Student/Teacher enter to the website |  |
|  | 1. System displays the form to registered |
| 1. Student/Teacher fill the form and enter finish |  |
|  | 1. System check the data |
|  | 1. If valid create new account and display success |
|  | 1. If not valid ask to re-enter the data |
| Exceptions: | **User Action** | **System Action** |
|  | 1. Web site is fail |
|  | 1. Try again to sign up |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Log in | |
| Actors: | Student , Teacher | |
| Pre-conditions: | Have a account | |
| Post-conditions: | Enter to his account | |
| Flow of events: | **User Action** | **System Action** |
| 1. User enter to the website |  |
|  | 1. System display the user as a student or teacher |
| 1. Choose if student or teacher |  |
| 1. Log in |  |
|  | 1. Enter to his account |
| Exceptions: | **User Action** | **System Action** |
| 1. Enter wrong user or password |  |
|  | 1. Display wrong log in |
|  | 1. Display message sign up or forget password |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Play game | |
| Actors: | Student | |
| Pre-conditions: | Log in to his account | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. System display category of games |
| 1. Student choose category of the games. |  |
|  | 1. System display games to the student |
| 1. Student choose the game he want |  |
| 1. after student finish the game click finish |  |
| Exceptions: | **User Action** | **System Action** |
| 1. user stop the game without log out |  |
|  | 1. system save the current score |
| Includes: | Save score | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Count a game score | |
| Actors: | System | |
| Pre-conditions: | Play game | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. System count the student’s score |
|  | 1. System display Scores of the student |
|  | 1. System save scores of the student in his account |
| Exceptions: | **User Action** | **System Action** |
|  | 1. System is crashed |
|  | 1. System save the last score |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Rating a game | |
| Actors: | Student | |
| Pre-conditions: | The student is play the game that he want to rate | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. System display if the student make , change or cancel the rate |
| 1. If Student choose to make a rate |  |
|  | 1. System display the games that want to rate |
| 1. Student choose the game |  |
|  | 1. System display the options of rating (interesting, normal, boring) |
| 1. Student choose the rating |  |
|  | 1. System save the rate |
|  | 1. If Student choose to change a rate |  |
|  |  | 1. System display the games that want to change a rate |
|  | 1. Student choose the game |  |
|  |  | 1. System display old rate |
|  |  | 1. System display the options of rating (interesting, normal, boring) |
|  | 1. Student choose new rating |  |
|  |  | 1. System save new rate |
|  | 1. If the Student choose to cancel a rate |  |
|  |  | 1. System display the games that want to change a rate |
|  | 1. Student choose the game |  |
|  |  | 1. System display the games that rated |
|  | 1. Student choose the game that he want to cancel the rate |  |
|  |  | 1. System remove rating |
|  |  | 1. System save a cancelation |
| Exceptions: | **User Action** | **System Action** |
|  | 1. If student enter the rate and the system is crashed |
|  | 1. System ask the student to rate the game again |
|  |  | 1. If student enter the new rate after changing and the system is crashed |
|  |  | 1. System save the last rate and ask for changing the rate again |
|  |  | 1. If student cancel the rate and the system is crashed |
|  |  | 1. System not cancel the rate and ask for cancelling rate again |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Commenting a game | |
| Actors: | Student | |
| Pre-conditions: | The student is play the game that he want to comment | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. System display if the student make , change or cancel a comment |
| 1. If Student choose to make a comment |  |
|  | 1. System display the games that want to comment |
| 1. Student choose the game |  |
|  | 1. System display the place of writing the comment |
| 1. Student Write Comment |  |
|  | 1. System save the Comment |
|  | 1. If Student choose to change a comment |  |
|  |  | 1. System display the games that want to change a comment |
|  | 1. Student choose the game |  |
|  |  | 1. System display old comment |
|  |  | 1. System display the place of new comment |
|  | 1. Student write a new comment |  |
|  |  | 1. System save new comment |
|  | 1. If the Student choose to cancel a comment |  |
|  |  | 1. System display the games that want to cancel a comment |
|  | 1. Student choose the game |  |
|  |  | 1. System remove a comment |
|  |  | 1. System save a cancelation |
| Exceptions: | **User Action** | **System Action** |
|  | 1. If student enter the comment and the system is crashed |
|  | 1. System ask the student to comment the game again |
|  |  | 1. If student enter the new comment after changing and the system is crashed |
|  |  | 1. System save the last comment and ask for changing the comment again |
|  |  | 1. If student cancel the comment and the system is crashed |
|  |  | 1. System not cancel the comment and ask for cancelling a comment again |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Create game | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher enter to his account | |
| Post-conditions: | Trying the game | |
| Flow of events: | **User Action** | **System Action** |
| 1. Teacher click to create a game. |  |
|  | 1. system displays the categories of games |
| 1. teacher choose the suitable category “Match pictures” , “Multiple Choices game” , “Run code game” and add the game |  |
| 1. teacher click finish |  |
|  | 5- the system save it and log out from the game |
| Exceptions: | **User Action** | **System Action** |
|  | 1. System is crashed |
|  | 1. System remove the game |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Try a game | |
| Actors: | Teacher | |
| Pre-conditions: | Created a game | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1. Teacher click to try to test the game |  |
|  | 1. .the system displays the games which created by teacher |
| 1. teacher choose the game he want to try if its run correct or not. |  |
| 1. If correct click finish |  |
|  | 1. the system save and log out from the game |
| 1. if the game not valid click to edit it |  |
|  | 1. the system edit it and save |
| Exceptions: | **User Action** | **System Action** |
|  | 1. System is crashed |
|  | 1. System reload the game |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Edit a game | |
| Actors: | Teacher | |
| Pre-conditions: | Created a game | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1. Teacher click edit to fix the game . |  |
|  | 1. the system displays the games which created by teacher |
| 1. teacher choose the game he want to edit |  |
| 1. teacher click finish |  |
|  | 5- the system save it and log out from the game |
| Exceptions: | **User Action** | **System Action** |
|  | 1. System is crashed |
|  | 1. System not make edit and ask to edit again |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Remove a game | |
| Actors: | Teacher | |
| Pre-conditions: | Created a game | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1. Teacher click to remove the game. |  |
|  | 1. the system displays the games which created by teacher |
| 1. teacher choose the game he want to remove |  |
| 1. teacher click finish |  |
|  | 5- the system remove it and save it |
| Exceptions: | **User Action** | **System Action** |
|  | 1. System is crashed |
|  | 1. System not make remove and ask to remove again |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | respond to student comment | |
| Actors: | Teacher | |
| Pre-conditions: | Student make comments | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1. Teacher click to view comments of game |  |
|  | 1. the system displays the games which created by teacher |
| 1. teacher choose the game he want to view student’s comments |  |
| 1. if teacher want to reply click to reply to the comment |  |
|  | 1. the system display a place to reply of comment |
| 1. teacher write his comment |  |
|  | 1. the system save it . |
| Exceptions: | **User Action** | **System Action** |
|  | 1. System is crashed |
|  | 1. System ask the teacher to reply comment to student for the game again |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Log out | |
| Actors: | Teacher/student | |
| Pre-conditions: | Log in to the website | |
| Post-conditions: |  | |
| Flow of events: | **User Action** | **System Action** |
| 1. Student/Teacher click log out |  |
|  | 1. the system save all changing and log out from the Account . |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Nonfunctional -> Performance ,Scalability ,Usability  Use cases -> play game , save score , rating , comment | *Abraam Emad* |
| Nonfunctional -> Reliability ,Security ,Maintainability  Use cases -> sign up , choose , log in. | *Ahmed Ibrahim Mokhtar* |
| Nonfunctional -> Implementation ,Interface ,Legal  Use cases -> create game , try game , edit game , remove game , respond to student game , log out | *Ali Mohamed EL-Amin* |