**Cairo University  
Faculty of Computers and Information** 

**CS251**

**Software Engineering I**

Project Name<Science Academy>

Software Requirements Specifications

Team Name<Mega Minds>

11/2016

Contents

[Team 3](#_Toc466140284)

[Document Purpose and Audience 3](#_Toc466140285)

[Introduction 3](#_Toc466140286)

[Software Purpose 3](#_Toc466140287)

[Software Scope 3](#_Toc466140288)

[Functional Requirements 4](#_Toc466140289)

[Non Functional Requirements 7](#_Toc466140290)

[System Model 9](#_Toc466140291)

[Use Case Model 9](#_Toc466140292)

[Use Case Tables 10](#_Toc466140293)

[Ownership Report 52](#_Toc466140294)

# 

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140292 | Noura Mohamed Arafa | nouraarafa95@gmail.com | 01113600147 |
| 20140313 | Yasser Mohamed Abdel Hameed | co2yasser@gmail.com | 01115658342 |
| 20140278 | Menna Allah Ali Mohamed | mennaali365@gmail.com | 01141155710 |

# Document Purpose and Audience

* This document mainly for the software developer, admins and project manager.
* This document mainly written to show the reader the scope of project, Use Cases diagrams and tables, purpose, functional and nonfunctional requirements.

# Introduction

## Software Purpose

The website will provide games that explain the scientific terms in easy way

## Software Scope

It’s a website not a mobile application or desktop application, you just need to have an internet source to get on.

The website have games with specific subjects and categories, teacher can make it and the student will play it.

## Functional Requirements

* **Sign up**

The user can sign up to the website by press ‘sign up’ button of the tool bar

* **Make game**

The teacher can create the game and he has the ability to

* choose the subject

He can choose the game’s subject from the subject list which specified by the admin

* choose game category

He can choose the game’s category from the category list which specified by the admin

* choose the scoring criteria

He can choose the game’s scoring criteria from the scoring criteria list which specified by the admin

* make a description

The teacher can also create programming game and he has the ability to

* choose languages

The system will have a default languages and the teacher will choose one of them

* determine number of levels
* enter content

The teacher will enter the content he wants

* create class

When the teacher create a programming game the system create by default a class which have “created game, sorted ranks, announcements, upcoming events”

* **My Profile**

Teacher/Student press on My Profile button from system tool bar, system will display lists of choices (view Profile and logout).

* **view profile**

Teacher/Student can view his profile to see his personal information and his image by choosing view profile from My Profile list.

* **change Image:**

Teacher/Student can change his personal image by pressing on it so system will make him choose a new one and then system will save the new image after teacher press on save button.

* **Edit profile**

Teacher/Student can edit his personal information if he presses on edit profile button so system enable him to make changes and then system will save them and teacher can delete his account if wanted.

* **View Achievements**

Teacher/Student can view his Achievements by pressing on View Achievements after entered his Profile, the system will display date of signup and his achievements in each game he has played, his highest score and level reached in each one and teacher can start to play any of them.

* **View My Classes**

Teacher choose view My Classes from system tool bar, the system will display list of the classes that the teacher created and teacher can choose any class of them to see the it's details which is (number of students registered in that class, average number of completed levels, average level play time, average rate and date of creation) and he can get the code of that class by pressing on get code button.

* **Remove class**

Teacher can remove his class by pressing on remove class button from class page.

* **edit class**

Teacher can edit his class by pressing on edit class button from class page.

* **View hided classes**

Teacher can see his hided classes by pressing on view hided classes' button and can unhide any of them

* **Play game**

Student or Teacher can play any selected game

* **Search for game**

Student or Teacher can search for game by name or by filtration.

* **Game rate and comment**

Student and teacher can rate or comment on any game.

* **Notification and reply**

Student and teacher can see notification and reply on comment.

* **Send message**

User can send several messages to another user in the website by open the chat between them

* **Receive message**

User can receive several messages from another user in the website by open the new message and can reply to it.

* **Play programming game**

Student can play a programming game which is in a specific class or in the main page of programming section.

* **Contest**

Student can enter a contest which may appear after specific levels and he can skip it, contest contain a problem solving.

* **Fight**

Student can enter a fight with another student, they will play an animation game about a specific concept of programming.

* **Make announcement**

Teacher can make announcement and the system publish it to the student.

* **Search for game**

Teacher or student can search for game by name or filtering.

* **View class**

Teacher can view his class and he can make his class private if he wants

## Non Functional Requirements

* Usability

Make a good GUI to make it easy to use.

* Supportability

If any issues happen to the website or user must be supported web site must be flexible.

* **Performance requirements**

The error messages will be display into 3 seconds

Remove the game will take 10 seconds

Edit the profiles will take 10 seconds

Remove the account will take 15 seconds

Flash the account will take 20 seconds

* **Data base**

We’ll use the data base in various situation like signup, remove account, edit profile, log in all this operation have the database properties (automaticity, durability, consistency)

* **Availability**

The web site available 24 by 7

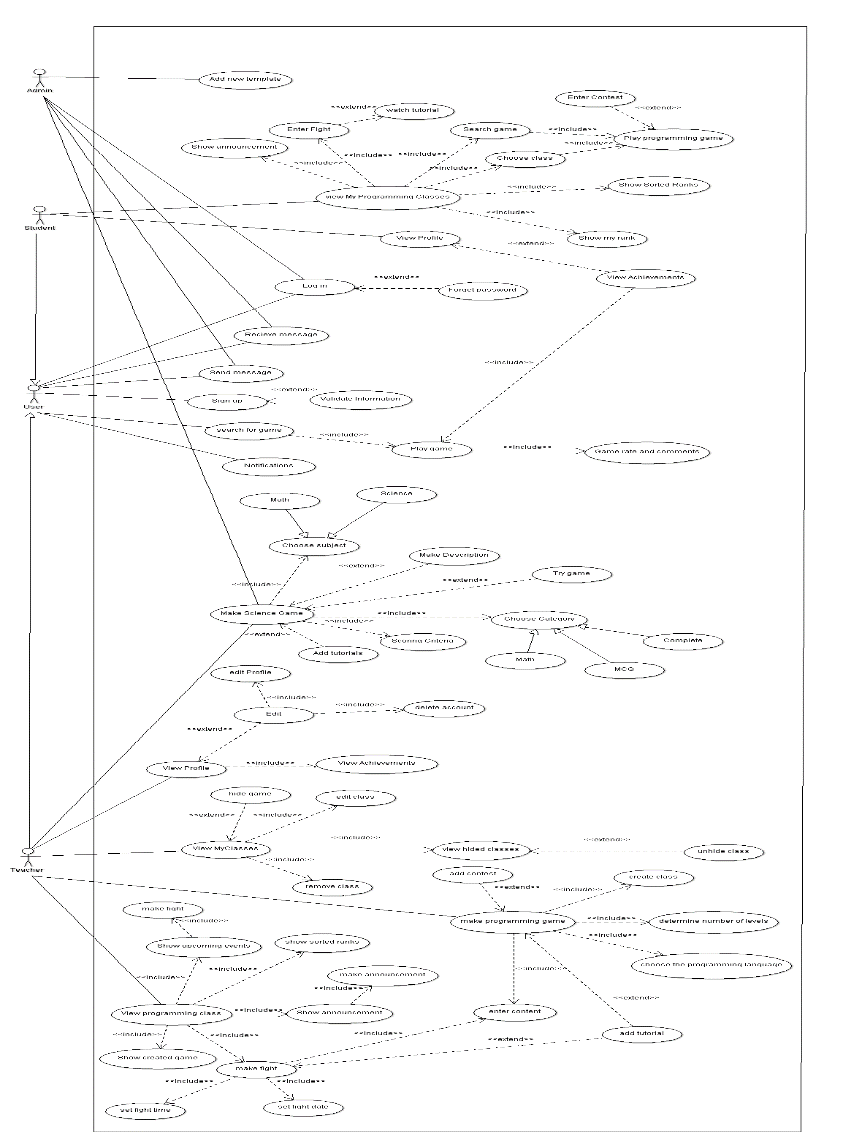
* **Security**

The website will save all the personal information in the database and no one can access this database

# 

# System Model

## Use Case Model



## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Sign up | |
| Actors: | Teachers or students | |
| Pre-conditions: | The user open the web page and want to make a registration | |
| Post-conditions: | The account will be created | |
| Flow of events: | **User Action** | **System Action** |
| 1-the user press “sign up” button |  |
|  | 2-system will show 2 options teacher or student |
| 3-the user will choose his type and press next |  |
|  | 4-the system save his type ,show the personal list to this type |
| 5-the user will fill the information by uses the use case validate |  |
|  | 6-system save the changes |
| 7-user press confirm for the terms of service then press finish |  |
|  | 8-the system save this account information into the data base then move the user to the main page |
| Exceptions: | **User Action** | **System Action** |
| 1-if the user didn’t choose type |  |
|  | 2-the system show the error message “you should enter a type” |
| 3-if he didn’t confirm the terms of service |  |
|  | 4-the system show the error message “you must confirm the terms of service” |
| Includes: | * Validate information   It will validate the personal information | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Log in. | |
| Actors: | Teacher/Student/Admin. | |
| Pre-conditions: | Open Website. | |
| Post-conditions: | Logged in the website. | |
| Flow of events: | **User Action** | **System Action** |
| 1- Teacher/Student/Admin input his username/email and his password. |  |
|  | 2- System will validate data and get his profile information and will display the main page. |
| 3-If Teacher/Student/Admin press on forget password. |  |
|  | 4-System will ask him to enter his email. |
| 5-User enter his email. |  |
|  | 6-System will validate the entered email then send a confirmation code to his email so he can change the password. |
|  | 7-System ask User to enter the confirmation code. |
| 8-User entered the code. |  |
|  | 9-system will validate it and then ask the user to enter a new password twice. |
| 10-User will enter the new password twice. |  |
|  | 11- System will get his profile information and will display the main page. |
| Exceptions: | **User Action** | **System Action** |
| 1-If input a wrong data. |  |
|  | 2-System will display "error username/email or password". |
| 3- If teacher entered an invalidate email will recovery his password. |  |
|  | 4-System will display "Error email re-enter it ". |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Validate information | |
| Actors: | Teachers or students | |
| Pre-conditions: | The user press the signup button and now will enter his personal information | |
| Post-conditions: | The information validated | |
| Flow of events: | **User Action** | **System Action** |
| 1-if the user enter user name |  |
|  | 2- system check if this name had been used before, then save it |
| 3.1-if the user enter email |  |
|  | 4.1-system will check if this a validation email, then sent a confirmation email |
| 3.1.1-the user confirm the verification email |  |
|  | 4.1.1-system save this email |
| 5-if the user enter the password |  |
|  | 6-system check the password matching the requirements then save it |
| 7-the user enter the other personal information |  |
|  | 8- the system save it |
| Exceptions: | **User Action** | **System Action** |
| 1-enetring repeated user name |  |
|  | 2-system show error message “it has been used before” and make him enter another one |
| 3-entering not valid email |  |
|  | 4- system show error message “it is invalid email” and make him to write a right email |
| 5-if he didn’t confirm the email |  |
|  | 6- system show error message “you should confirm the email” |
| 7-if he entered a password which not matching the requirements |  |
|  | 8- system show error message” please follow the requirements” |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Play game. | |
| Actors: | Teacher, students and admins. | |
| Pre-conditions: | Login and game selected to play. | |
| Post-conditions: | User see the score | |
| Flow of events: | **User Action** | **System Action** |
| 1- User read game roles. |  |
| 2- user start the game |  |
|  | 3- System send questions for player. |
| 4- User answer the questions and submit answers |  |
|  | 5-Referee check the criteria and send the score based on the criteria. |
| 6-User get and see the score. |  |
| Exceptions: | **User Action** | **System Action** |
| 1-if the user press “back” |  |
|  | 2-the system show a message “do you want to go out and save the level” |
| 3.1-if the user press “yes “ |  |
|  | 4.1-the system save the point that the student stopped and check the criteria and send the score based on the criteria and return him to the previous page |
| 3.2-if the user enter “no” |  |
|  | 4.2-the system won’t move the user to any page |
| Includes: | Referee and game selection | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Game rate and comments | |
| Actors: | Teacher, students | |
| Pre-conditions: | After playing game or selected game | |
| Post-conditions: | Show to user successful rating or comment | |
| Flow of events: | **User Action** | **System Action** |
| 1- User may rate the game and may comment on it and then press on submit button. |  |
|  | 2- System get data from user and save in the DB and send notification to the game owner. |
|  | 3- System view to the user a successful message. |
| Exceptions: | **User Action** | **System Action** |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Make science game | |
| Actors: | Teachers | |
| Pre-conditions: | The teacher login and want to make a game | |
| Post-conditions: | The game had been created and added to the teacher classes | |
| Flow of events: | **User Action** | **System Action** |
| 1-The teacher press ”make science game” button |  |
|  | 2-system show subject list |
| 3- Teacher choose the subject from the list. |  |
|  | 4- System save the teacher choice then show the category list |
| 5-Teacher uses the use case choose category. |  |
|  | 6-system save the teacher choice then show the template of scoring criteria |
| 9-teacher full the scoring criteria template. |  |
|  | 10-system save the template then show the description text box |
| 11.1-if the teacher want to write a description he will fill the text box |  |
| 11.2-if the teacher doesn’t want to write the description he will press skip button |  |
|  | 12-system save the changes |
| 13-the teacher will add the tutorial to each level of the game and if he want to add hints in some levels |  |
|  | 14-the system save the changes and show the game page which have the description and the game. |
| 13-he’ll press finish button. |  |
|  | 14-the system save the game in the teacher classes and the created games list. |
| Exceptions: | **User Action** | **System Action** |
| 1- The teacher didn’t full the category template details. |  |
|  | 2- the system show an error message “you must complete the fields “ |
| 3-the teacher didn’t full the scoring criteria template. |  |
|  | 4-the system show an error message “you must full the criteria template “ |
| 5-the teacher press logout button |  |
|  | 6-the system show a message “Are you sure you don’t want to make this game” |
| Includes: | * Choose subject   The teacher will choose the subject   * Choose category   The teacher will choose the game category and full the templates   * Filling the scoring criteria   The teacher will set the criteria for the scoring counters and the levels limits | |
| Notes and Issues: | The subjects and categories can be extend | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Choose category | |
| Actors: | Teachers | |
| Pre-conditions: | The teacher login, press make a game ,chose the subject and the system showing to him the category list | |
| Post-conditions: | The category will be chosen | |
| Flow of events: | **User Action** | **System Action** |
| 1-if the teacher choose any subject |  |
|  | 2-system save the choice and show the category list” Match, MCQ, Complete”. |
| 3-the teacher choose his choice from the category list |  |
|  | 4-system save the changes |
| Exceptions: | **User Action** | **System Action** |
| 1-if the teacher didn’t choose any category |  |
|  | 2-system show error message “you must choose category” |
| 3-if he didn’t set programming language |  |
|  | 4-system show error message “you must set programming language” |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Make Programming game | |
| Actors: | Teachers | |
| Pre-conditions: | The teacher want to Make programming game | |
| Post-conditions: | The programming game will be done | |
| Flow of events: | **User Action** | **System Action** |
| 1-the teacher enter the programming section and choose “make programming game “button from the tool par. |  |
|  | 2-the system show a text box for the class name and ID |
| 3-the teacher enter the class name and ID. |  |
|  | 3-system save the changes and create a class contain the (created game ,ranks ,announcements ,upcoming events “fights”) and then show the a template for the content |
| 5- the teacher choose the content and press “next” |  |
|  | 6-system save the content and show the number of levels box |
| 7-teacher enter the number of levels and press “next” |  |
|  | 8-system save the changes and show the programming language box |
| 9-teacher chooses the language and press “next” |  |
|  | 10-system saves the changes and allow the user to put the score criteria |
| 11-the teacher enter the score criteria |  |
|  | 12-system save changes then show the message “are you want to add tutorials” |
| 13.1-if the teacher choose “yes” |  |
|  | 14.1-the system move the teacher to blank page |
| 13.2-the teacher add tutorials and hints if he want and press “next” |  |
| 13.3-if the teacher use NO |  |
|  | 14.2 – the system show a message “are you want to add contests” |
| 15.1-if the teacher press “yes” |  |
|  | 16.1-the system ask the teacher to enter the contest place between the levels |
| 15.2-the teacher enter the contest place and press next |  |
|  | 16.2-the system save changes and allow the teacher to enter the content |
| 15.3-the teacher enter the content and press finish |  |
|  | 16.3-the system save the changes and arrange the game’s levels with the added contests |
| 15.4-the teacher choose No |  |
|  | 17-the system save the game to the created game container into the teacher class, and show the main page of the game. |
| Exceptions: | **User Action** | **System Action** |
| 1-if the teacher press “cancels” in entering the content, choosing the number of levels and language and adding the tutorials. |  |
|  | 2-the system show a message “Are you sure you don’t make the game” |
| 3.1-if the user choose “yes” |  |
|  | 4.1- system cancel this operation |
| 3.2-if the user choose “No” |  |
|  | 4.2- the system return him to the last stopped stage of making the game |
| Includes: | Choose content, determine number of levels, choose language, add tutorial, create class | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Make fight | |
| Actors: | Teachers | |
| Pre-conditions: | The teacher want to make fight | |
| Post-conditions: | The fight finished | |
| Flow of events: | **User Action** | **System Action** |
| 1-the teacher choose “make fight“ |  |
|  | 2-system show the a template for the content |
| 3- the teacher enter the content and press “next” |  |
|  | 4-system save the content |
| 5-teacher enter the tutorials and press “next” |  |
|  | 6-system save the changes and show message “set the fight time” |
| 7-teacher set the time and press “next” |  |
|  | 8-system save the changes and show message “set the fight date” |
| 9-the teacher set the fight date and press “next” |  |
|  | 10-the system save the changes and make the user to enter the ranking criteria |
| 11-the teacher enter the ranking criteria. |  |
|  | 12- the system save the changes and show the announcement for the fight box |
| 13-the teacher enter the announcement and press “finish” |  |
|  | 14-the system save the changes and save the announcements in the announcements container into the teacher class then upload it into the class home page |
| Exceptions: | **User Action** | **System Action** |
| 1-if the teacher press “cancels” in entering the content, entering the tutorial, set the fight date, set the fight time. |  |
|  | 2-the system show a message “Are you sure you don’t make the fight” |
| 3.1-if the user choose “yes” |  |
|  | 4.1- system cancel this operation |
| 3.2-if the user choose “No” |  |
|  | 4.2- the system return him to the last stopped stage of making the game |
| 5-if the teacher press “next” in entering the content, entering the tutorial, set the fight date, set the fight time without filling the data |  |
|  | 6-the system show the message “you must fill the empty field” |
| Includes: | entering the content, entering the tutorial, set the fight date, set the fight time | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Make announcements | |
| Actors: | Teachers | |
| Pre-conditions: | Log in and entered the class. | |
| Post-conditions: | The announcements uploaded into the class | |
| Flow of events: | **User Action** | **System Action** |
| 1-the teacher choose “make announcements“ |  |
|  | 2-system show the announcements textbox |
| 3- the teacher enter the announcements and press “finish” |  |
|  | 4- the system save the announcements in the announcements container into the teacher class ,send a notification to the class students , then upload it into the class home page |
| Exceptions: | **User Action** | **System Action** |
| 1-if the teacher press “cancels” in entering the announcements. |  |
|  | 2-the system show a message “Are you sure you don’t make the announcements” |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | View Profile | |
| Actors: | Teacher/Student | |
| Pre-conditions: | Teacher/Student log in and press on My Profile button from tool bar. | |
| Post-conditions: | Teacher/Student viewed his profile OR edited it. | |
|  | 1-If Teacher/Student press on view Profile button from My Profile list. |  |
|  | 2-System display Teacher's profile which contain personal information and his Image and view Achievements button and edit profile button. |
| 3-If Teacher/Student press on his Image. |  |
|  | 4-Sysetm will ask teacher choose image from his PC. |
| 4-If Teacher/Student choose an image from his PC. |  |
|  | 5-System will display the image, update image button and cancel button. |
| 5.1-If Teacher/Student press on update image button. |  |
|  | 5.1.1-System will update the image and save it. |
| 5.2-If Teacher/Student press on cancel button. |  |
|  | 5.2.1-The system will return old image and go back to view profile page. |
| 6-If Teacher/student make change in any information and entered his password in the text box which is under his information in order to save changes and then press on save button. |  |
|  | 7- System will validate the entered password and if it is correct, it will validate the new inputs then save the changes into the database. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher/Student press on back button |  |
|  | 2- System return to previous page. |
| 3-If Teacher/Student press on log out button from My Profile list. |  |
|  | 4-System will close his profile and go to log in page. |
| 5-If Teacher/Student make any change either input new information or delete his information and don't press save but press on back button or log out. |  |
|  | 6-System will discard changes and obey his request. |
| 7-If Teacher/Student entered a wrong password Or press save without entered the password. |  |
|  | 8-system will display "Incorrect password". |
| Includes: | **1-**If the Teacher/Student use the use case delete account he can delete his account.  **2-**By using use case view Achievements he can see what he reached in each game he had played and highest Score in each game. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | View Achievements. | |
| Actors: | Teacher/Student | |
| Pre-conditions: | Teacher/Student choose View profile and then press View Achievements button. | |
| Post-conditions: | Teacher/Student viewed his Achievements OR start playing a game. | |
|  | 3- Teacher/Student press on view Achievements button from his Profile page. |  |
|  | 4- System display date of sign up, list of games he had played and the system will display on each game the date of last played time, level he reached and the highest score he achieved in that game. |
| 5-If Teacher/Student press on continue playing button of a specific game he can use play game use case. |  |
|  | 6-System will start the game. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher/Student press on back button |  |
|  | 2- System return to previous page. |
| 3-If Teacher/Student press on log out button from My Profile list. |  |
|  | 4-System will close his profile and go to log in page. |
| Includes: | * Play game   He’ll play any created game | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 13 | |
| Use Case Name: | Edit profile. | |
| Actors: | Teacher/Student | |
| Pre-conditions: | Teacher/Student choose view profile and press on edit button from view Profile page. | |
| Post-conditions: | Teacher/Student edited his profile. | |
|  | 1- Teacher/Student choose edit profile from view Profile page. |  |
|  | 2- System display his personal information. |
| 3-If Teacher/Student make any change in his personal information. |  |
|  | 4-System will display the save button. |
| 5-If Teacher/Student press save. |  |
|  | 6-. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher/Student make change in his information and don't press save and press on back button or log out button |  |
|  | 2- System will display a message "Are you want discard changes?!" |
| 2.1-If Teacher/Student press yes, Discard. |  |
|  | 2.1.1-system will discard the change and system will get him back to previous page/log out to welcome page. |
| 2.2-If Teacher/Student press No, Save. |  |
|  | 2.2.1- System will validate the new inputs and then save the changes. |
| 3-If Teacher/Student entered a wrong password. |  |
|  | 4-System will display an error message "You entered a wrong password ". |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 14 | |
| Use Case Name: | Delete account. | |
| Actors: | Teacher/Student | |
| Pre-conditions: | Teacher/Student choose view profile and press on delete account button from view Profile page. | |
| Post-conditions: | Teacher/Student delete his account. | |
|  | 1-If Teacher/Student press delete account. |  |
|  | 2-System ask the teacher to enter his email/user name and password. |
| 2.1- Teacher/Student entered his email/user name and his password and then press confirm deletion. |  |
|  | 2.1.1- System will validate that inputs then send a confirm email to his email which is in the data base and after confirming the email the system will delete his account from data base and put it in deleted account data base. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher/Student make don't press confirm deletion and press on back button or log out button |  |
|  | 2- System will display a message "Are you don't want to delete your account?!" |
| 2.1-If Teacher/Studentpress yes, don't delete. |  |
|  | 2.1.1- System will get him back to previous page/log out to welcome page. |
| 2.2-If teacher press No, delete. |  |
|  | 2.2.1-System will continue his process. |
| 3-If Teacher/Student entered a wrong password. |  |
|  | 4-System will display an error message "You entered a wrong password ". |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 15 | |
| Use Case Name: | View My Classes. | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher log in. | |
| Post-conditions: | Teacher viewed, edited or deleted any class he wanted. | |
|  | 1-Teacher press on My Classes from system tool bar. |  |
|  | 2- System display list of his Classes. |
| 3-If Teacher choose any class from Classes list. |  |
|  | 4-System will display number of students registered in that class, average number of completed levels, average level paly time, average rate and date of creation. |
| 5-If Teacher press get class code. |  |
|  | 6-System will return game code. |
| 7- If Teacher press on hid class button. |  |
|  | 8-System will make this class private and hide continue playing button from student's list. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher press on back button. |  |
|  | 2- System return to the previous page. |
| Includes: | **1-**By using use case remove class he can remove this class.  **2-**By using use case edit class teacher can edit this class.  **3-**By using use case view hided classes teacher can see the hided classes if exist. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 16 | |
| Use Case Name: | Remove Class. | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher choose class from classes list and press Remove class. | |
| Post-conditions: | Class is removed. | |
|  | 1-If Teacher press on remove button that is inside class page. |  |
|  | 2- System Ask teacher to input password of his account. |
| 3-If Teacher input his password and press on delete. |  |
|  | 4-System validate password and if password is correct system will delete the game and hid continue playing button in that Class from student's list. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher entered wrong password. |  |
|  | 2- System show an error message "You entered wrong password try again". |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 17 | |
| Use Case Name: | Edit Class. | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher choose class from classes list and press edit class. | |
| Post-conditions: | Class is edited. | |
|  | 1-If Teacher press on edit game button that is inside class page. |  |
|  | 2- System display class name, class description, scoring criteria and add new level button. |
| 3- If Teacher make changes in any information that the system displayed and press on edit button. |  |
|  | 4-System will display confirmation message "Are you sure you want to edit this information ". |
| 4.1-If Teacher press confirm. |  |
|  | 4.1.1-System will update changes to that class. |
| 4.2-If Teacher press cancel button. |  |
|  | 4.2.1-System will stile in editing page. |
| 7-If teacher press on add new level button. |  |
|  | 8-The system will ask him to enter score range of that level and fill templates. |
| 9-If teacher entered them and press Done button. |  |
|  | 10-The system will update changes to that class. |
| Exceptions: | **User Action** | **System Action** |
| 1- If Teacher press on back button or log out without saving changes. |  |
|  | 2- The system show an error message "Are you want to discard changes?!" |
| 2.1-If teacher press No, save changes. |  |
|  | 2.1.1-System will save changes and update them to data base. |
| 2.2-If teacher press Discard. |  |
|  | 2.2.1-System will discard changes and go to previous page. |
| 5-If teacher after press add new Level and press back button/ log out and don't fill required data |  |
|  | 6-The system will display a message to alter teacher "Are you want to cancel request?" |
|  |  |
| 6.1-If teacher press No, complete. |  |
|  | 6.1.1-system will display add new level page. |
| 6.2-If teacher press yes, cancel. |  |
|  | 6.2.1-system will cancel and go back to edit page. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 18 | |
| Use Case Name: | View hided classes. | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher press View My Classes button. | |
| Post-conditions: | Teacher view his hided classes. | |
|  | 1-If Teacher press View hided classes' button that is below View My Classes title. |  |
|  | 2- System display hided classes and date of hiding them. |
| 3-If teacher press on Unhide |  |
|  | 4-System make it visible and return continue playing button in Student's classes list. |
| Exceptions: | **User Action** | **System Action** |
| Includes: |  | |
| Notes and Issues: |  | |

# 

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 20 | |
| Use Case Name: | view programming class | |
| Actors: | Teachers | |
| Pre-conditions: | The teacher login ,and he want to view class | |
| Post-conditions: | The teacher saw the created game, sorted ranks, announcements, upcoming events. | |
| Flow of events: | **User Action** | **System Action** |
| 1-the teacher choose “view class “ |  |
|  | 2-system show the class home page |
| 3.1- if the teacher press into the game name. |  |
|  | 4.1-the system move the user to the game home page |
| 3.1.1-if the teacher want to play the game he will use the use case” play game” |  |
| 3.1.2-if the teacher want to make a game he will us the use case table “make game” |  |
| 3.2-if the teacher want to see the ranks he will press into the” sorted ranks” |  |
|  | 4.2-the system show the score board with sorted ranks |
| 3.3-if the user want to see the announcements he will press “announcements” button |  |
|  | 4.3-the system move the user to the announcements page |
| 3.3.1-if the teacher want to make an announcements he will use the use case “make announcements” |  |
| 3.4-if the user want to see the upcoming events ,he will press “upcoming events” |  |
|  | 4.4-the system will move the user to the upcoming events |
| 3.4.1-if he want to make fight, he will use the make fight use case table |  |
| Exceptions: | **User Action** | **System Action** |
| 1-if the teacher press “back” |  |
|  | 2-the system will move the teacher to the previous page |
| Includes: | Use case make announcement, make fight, show upcoming events, show announcements . | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 21 | |
| Use Case Name: | View My programming class | |
| Actors: | Students. | |
| Pre-conditions: | Log in and press on the programming section from system tool bar. | |
| Post-conditions: | Viewed his class. | |
| Flow of events: | **User Action** | **System Action** |
| 1.1-If Student search about class by name or ID. |  |
| 1.2-If student press “My classes” button |  |
|  | 1.2.1-System will display all classes which he registered in. |
| 1.2.2-Student choose a class. |  |
|  | 2-system will display the page of that class which contain(show announcement, show sorted ranks, My rank, upcoming events and play) buttons . |
| 3.1-If student press on show announcement button. |  |
|  | 4.1-System will display any announcement about that class. |
| 3.2-If student press on show sorted ranks button. |  |
|  | 4.2-System will display the sorted ranks of that class. |
| 3.3-If student press on My rank button. |  |
|  | 4.3-System will display his rank on that class. |
| 3.4-Student can use “use case” upcoming events by press on upcoming events button. |  |
| 3.5-Student can use “use case” play by press on play button. |  |
| Exceptions: | **User Action** | **System Action** |
|  |  |
| Includes: | Use case Play game.  Use case upcoming events.  Use case show announcement.  Use case show ranks.  Use case My Rank. | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 22 | |
| Use Case Name: | Play programming game. | |
| Actors: | students | |
| Pre-conditions: | Log in and choose class from view classes page. | |
| Post-conditions: | Played a game. | |
| Flow of events: | **User Action** | **System Action** |
| 1-If student press on “play game “ button. |  |
|  | 2.1-if it’s the first time to enter the game the system show the role’s game. |
| 3.1-the user read the game’s roles and press “skip” |  |
|  | 2.1.1 the tutorial of the level appear |
| 3.2 the user will understand the tutorial and press “skip” |  |
| 3.3-User will start the game. |  |
|  | 2.2-if it isn’t the first time the system get the last level ”the last solution point” |
|  | 2.3-1if it’s the level’s begin ,the tutorial will appear |
| 4-User watch tutorial. |  |
| 5-User start the game. |  |
| 6-User can use “use case” contest if a contest exist. |  |
|  | 7-System calculate score after each level according to the criteria of the game |
| 8- If student press on hint button if exist. |  |
|  | 9-System will display the hint. |
| 10-If student press on exist the hint. |  |
|  | 11-System will return him to continue where he stop on the game. |
| Exceptions: | **User Action** | **System Action** |
|  |  |
| Includes: | Use Case Contest. | |
| Notes and Issues: |  | |

# 

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 23 | |
| Use Case Name: | Enter Contest | |
| Actors: | Student. | |
| Pre-conditions: | Playing a game and finish a specific levels. | |
| Post-conditions: | Entered a contest. | |
| Flow of events: | **User Action** | **System Action** |
| 1-If student finish a specific levels and a contest is appear to him so he can entered or skip it. |  |
| 2.1-If student press ”skip”. |  |
|  | 2.1.1-System will skip the contest and continue in the below level. |
| 2.2-If Teacher press on “enter”. |  |
|  | 2.2.1-System will open the contest which contain problems. |
| 3-Student will choose a problem to solve. |  |
|  | 4-System will display the content of the problem. |
| 5-If the student submit the solution by pressing on the submit button. |  |
|  | 6- System will check the answer and calculate rank and return it to the student. |
| 7-If the student press on ”finish” button. |  |
|  | 8-System will return him to continue the game. |
| Exceptions: | **User Action** | **System Action** |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 24 | |
| Use Case Name: | Enter Fight. | |
| Actors: | Students. | |
| Pre-conditions: | Select the class and press on upcoming event button. | |
| Post-conditions: | Make a fight. | |
| Flow of events: | **User Action** | **System Action** |
| 1-If the Student press on the current fight. |  |
|  | 2-System will open the fight which is an animation game. |
| 3-Student Start the fight. |  |
|  | 4-System calculate the rank and update it. |
|  | 5-After the game finished the system will display the winner and the sorted ranks. |
| 7-After finishing the fight if there is a tutorial Student can watch it. |  |
| Exceptions: | **User Action** | **System Action** |
| 1-If Student press on back button or log out. |  |
|  | 2-System will ask him “Are you want to surrender?” |
| 3-If student press “yes”. |  |
|  | 4-System will check the number of fighters and then make his request. |
|  | 4.1- if it is only one fighter it will be the winner. |
|  | 4-2-If it is more than one fighter they will continue the fight. |
| 5-If he press on “No”. |  |
|  | 6-System will make him continue the fight. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 25 | |
| Use Case Name: | add new templates | |
| Actors: | Admin | |
| Pre-conditions: | login admin mode | |
| Post-conditions: | templet is added | |
| Flow of events: | **User Action** | **System Action** |
| 1-admin insert new templet |  |
|  | 2-system get request from admin |
|  | 3-system saves the new templet |
|  | 4-system send success message |
| 5-admin see success message |  |
| Exceptions: | **User Action** | **System Action** |
| 1-press cancel |  |
|  | 2-system send message to save game |
| 3-admin send confirmation |  |
|  | 4- if yes the system save game, if no  System just cancel. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 26 | |
| Use Case Name: | Search for a game | |
| Actors: | Teacher, students and admins | |
| Pre-conditions: | Login | |
| Post-conditions: | show results | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enter id or name of game or filtering the games and press search. |  |
|  | 2.1- System send the game page to the user. |
|  | 2.2-if the system didn’t find it ,it’ll send a not found message |
| Exceptions: | **User Action** | **System Action** |
|  |  |
|  |  |
| Includes: |  | |
| Notes and Issues: |  | |

# Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| * From Use Case ID 4, 25,26 * Helping in Functional and nonfunctional Requirement. * Document purpose and Audience. | *Yasser Mohamed Abdel Hameed* |
| * Use Case ID 1, 3 and from 5 to 10, 18 * Helping in Functional and nonfunctional Requirement. * Purpose and scope. | Menna Allah Ali Mohamed |
| * From Use Case ID 11 to 17 & 19-24 * made Use Case Diagram. * Helping In Functional Requirement. | Noura Mohamed Arafa. |

# 