Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

CleverZone

Software Requirements Specifications

Nour , Mazen ,Mahmoud,Mahamed

November , 2016

Contents

[Team 2](#_Toc402452669)

[Document Purpose and Audience](#_Toc402452670) 2

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope](#_Toc402452673) 3

[Requirements 4](#_Toc402452675)

[Functional Requirements 4](#_Toc402452676)

[Non Functional Requirements 6](#_Toc402452677)

[System Models 7](#_Toc402452678)

[Use Case Model 7](#_Toc402452679)

[Use Case Tables 8](#_Toc402452680)

[Ownership Report](#_Toc402452681) 24

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140291 | Nour Ahmed Abbas | nourahmed@stud.fci-cu.edu.eg | 01211196825 |
| 20140245 | Mahmoud Ahmed Mahmoud | ouka.ahmed2@gmail.com | 01025517515 |
| 20140225 | Mohamed Ragab Mohamed | mohamed225.mr@gmail.com | 01096023385 |
| 20140211 | Mazen Mohamed Sayed | mazen.mohamed.967@gmail.com | 01009066848 |

# Document Purpose and Audience

* This is a Requirements Analysis Document. The document completely describes the system's functional, nonfunctional requirements, scenarios, use-cases and eradicates the conflicts between the client and the developers.
* The audience for this document include the users (Teacher, Student), client, the system developers, Project management team and the system designers.

# Introduction

## Software Purpose:

* Web based educational system target to make easy communication between the teacher and the student for easy and fun way for learning educational fields using games.

## Software Scope:

* The website severs two types of users with different capabilities
  + The teacher can create, delete and edit his\her games also he can create class rooms and play games .
  + The student can play games, enter class rooms of certain teacher, and has his's\her's score and total score in each class room.
* The website offers playing or creating games according to premade templates in three different categories (match picture, multiple choice game, and run code game) for the teacher.

# Requirements:

## Functional Requirements:

1. Register new account
2. Login
3. Logout
4. View profile.
5. Edit profile
6. Play game.
7. Create game.
8. Edit game.
9. Delete game.
10. Rate a game
11. Commenting on games.
12. Create class room
13. Delete class room
14. Add games to the class room
15. Delete games from the class room
16. Add student(s) to class room
17. Remove student(s) from class room
18. Share achievements with friends on different social networks.
19. Add game to favorite list
20. Add friends
21. Remove friends
22. Recommend friends
23. Join class chat room
24. Leave class chat room
25. View class room
26. View user profile
27. Search
28. Show friend’s score
29. Compete friend
30. Show Hall Of Fame List for every game category
31. View Home

**1 - Register new account for both teachers and students:**

The user will select to register as teacher or as student, then the user will enter the needed information (name, age, gender, email) the data will be validated and this email will be authenticated.

**2- Login:**

The user will select Login user will enter his\her email and his\her password and then this data will be validated.

**3** – **Logout:**

The user will select logout to end the current session.

**4- View Profile:**

The user will select my profile, it will show him\her all his informations, achievements, favorite games, followers list.

**5- Edit Profile:**

The user will select my profile, then he will be able to edit his/her information.

**6- Play Game:**

The user will select play game, then click the category, then select subject, then chooses the game. And start a game after that provide a game interface and controls according game templates. When the player wins or loses his\her score will be calculated then it will be added to his\her total score.

**7- Create game:**

The teacher will click my games, choose create game, then select the game category, he will enter the information (i.e. name, subject, age recommendation…),

And the game information according to the category then click publish game

**8-Edit game:**

The teacher will click my games, select the game he\she wants to edit, and then he\she will be able to change the information of the game.

**9-Delete game:**

The teacher will click my games; select the game he\she wants to delete.

**10 – Rate a game:**

After playing the game the player will be able to rate.

**11- Comment on Game:**

After playing the game & ratting it, the player will be able to rate.

12- Create Class room:

Teacher will be able to create class room and students to it

13- Delete class room

Teacher will be able to delete his own class room at any time.

14- Add games to the class room

Teacher will be able to add games to his own class room at any time

15- Remove games from the class room

Teacher will be able to remove games from his own class room at any time

16- Add students to the class room

Teacher will be able to add students to his own class room at any time

17- Remove students from the class room

Teacher will be able to remove students from his own class room at any time

18- Share achievements with friends on social networks

Student can share his/her achievements on social media like Facebook, Instagram.

19- Add a game to the favorite list.

Player (student or teacher) can add a new game to their favorite list for an easy access.

20 – Add friends

User can have friends in the friend list , so he \she can know his news in the home .

21 – Remove friend

User can remove anyone from this friends list

22 –Recommend friends

The system will recommend friends to the user according probabilistic analysis

23 – Open class room chat

The user can open the chat room in the class room.

24 –Leave the class room chat

The user can leave the chat room in the class room.

25 – view class room

The user can view the games and the chat of the class room.

26- View user profile

The user can view the other users profile.

27-Search

User can use search function to find someone/game by name.

28- Show friend’s score

For each game / category, player can show his friends score in it sorted descending order.

29- Compete friend

Player can compete with another player by inviting him.

30- Show hall of Fame List for every game category

Player can view the top-100 in a specific category.

31- View home

User can view home page to know the new update occurred on the website including his friends new update , created games from his followed teachers .

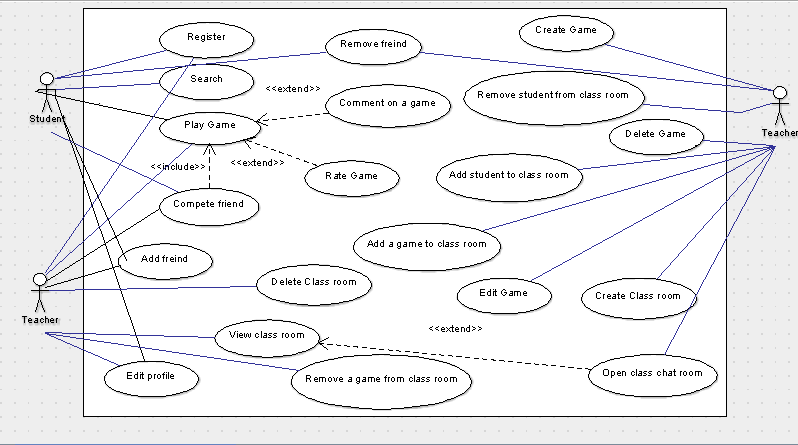
Non Functional Requirements :

|  |  |
| --- | --- |
|  | **Details** |
| **Performance** | * **Response time < 0.5 second to support interactive playing.** |
| **Security** | * **Users must register by using a valid e-mail address before they can use the website’s functions.** * **All Users’ personal data is secured and encrypted.** |
| **Usability** | * **Easy to play: Games starts with tutorials to get the users to be adapted to game controls and rules.** |
| **Supportability** | * **The website supports different subjects such as Math, Science, Technology, etc…** * **Different difficulty levels which suits all types of users and supports their learning process.** |
| **Interface** | * **The website offers different user interfaces for each type of user (Teacher, Student, Unregistered user) which contains the different functions they need to use the website.** |
| **Availability** | * **The maximum down time of the system is 15 minutes per week.** * **System have a backup on a secondary server.** |

# 

# System Models

## Use Case Model :



## Use Case Tables :

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Register | |
| Actors: | Site new user | |
| Pre-conditions: | The website opened | |
| Post-conditions: | The User will have a new account | |
| Flow of events: | **User Action** | **System Action** |
| 1 – the user will choose Student Registration or Teacher Registration |  |
|  | 2 – system will display The chosen registration form |
| 3 – the user enter The Data |  |
|  | 4 – The System will validate this data |
|  |  | 5 - System will create a new account |
| Exceptions: | **User Action** | **System Action** |
| 1- the entered data is not valid |  |
|  | 2- The system will inform the new user to enter the data correctly |
|  | 3-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 4- System ends the current login session and displays the main interface. |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Edit Profile | |
| Actors: | Teacher , Student | |
| Pre-conditions: | Login , view profile | |
| Post-conditions: | Update Profile information | |
| Flow of events: | **User Action** | **System Action** |
| 1- user Select profile tab |  |
|  | 2- System will display the profile. |
| 3 – the user select edit tab |  |
|  | 4- system will give a permission to edit the information |
| 5- user will edit the information and select save button |  |
|  | 6 – system will validate the updated information |
|  |  | 7 – system will update the information to the server |
|  |  | 8-system will display the profile |
| Exceptions: | **User Action** | **System Action** |
| 1- the user didn't select edit tab |  |
|  | 2- interface stay as it unless the user clicked any other tab in the interface |
|  | 3 – the user didn't select save |  |
|  |  | 4- the entered data will be neglected |
|  | 5 – the validation function return false on the entered data |  |
|  |  | 6 - the entered data will be neglected |
|  | 7-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 8- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Play Game | |
| Actors: | Teacher , Student | |
| Pre-conditions: | Login | |
| Post-conditions: | View Score of the game | |
| Flow of events: | **User Action** | **System Action** |
| 1- user Select play game tab |  |
|  | 2- system will display the Subjects list . |
| 3 – user will choose the subjects |  |
|  | 4- system will display the games  Categories |
| 5- user will choose the category |  |
|  | 6 – system will display the games |
|  | 7-user will choose the game |  |
|  |  | 8-system will display the game |
|  | 9-user will play the game |  |
|  | 10 – after the end of the game |  |
|  |  | 11- the system will prompt result message and the score , increase the total score of the user , show the rate and comment page |
|  |  | 12 – the system prompt two choices  Play again , back to all games |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Create Game | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | Game is created. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “My Games” tab, from the toolbar. |  |
|  | 2-The System will display this teacher's games page. |
| 3-User click “Create New Game” button |  |
|  | 4-System prompts a new interface which contain tools the new game. |
| 5- user will select the category he want and make the game according to the template in the interface |  |
| 6 – user will click save button |  |
|  | 5-System prompts a message box to confirm save edition. |
| 6- User confirms creation process by pressing “Save” button which exist in the message box. |  |
|  | 7- System creates the game, and prompts a message to confirm that creation process done successfully, the returns to “My Games” tab. |
| Exceptions: | **User Action** | **System Action** |
| 1. If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user didn't click save button |  |
|  |  | 4- the created game will be neglected |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Edit Game | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | Game is edited. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User selects the “My game” tab |  |
|  | 1. The system will display the teacher's games page. |
| 1. The teacher press on edit bottom on the right of the game name |  |
|  | 4-System display interface contains the tools to edit the game. |
| 5 – user will edit the game and click save |  |
|  | 6-System prompts a message box to confirm save edition. |
| 7- User confirms edition process by pressing “Save” button which exist in the message box. |  |
|  | 8- System edition the game, and prompts a message to confirm that edition process done successfully. |
| Exceptions: | **User Action** | **System Action** |
| 1. If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3-the user didn't click save |  |
|  |  | 4-the system will neglect the  change |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Delete Game | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher , Game already exist | |
| Post-conditions: | Game is deleted. | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “My Games” tab. |  |
|  | 1. System will display teacher's games page |
| 1. User click on delete button in the right of the game he wants to delete |  |
|  | 4- System prompts a message box to confirm deletion process. |
| 5- User confirms deletion process by pressing “OK” button which exist in the message box. |  |
|  | 5- System deletes the game, and prompts a message to confirm that deletion process done successfully |
| 6- the user press on confirm bottom. |  |
|  | 7- the game is deleted and the system returns to the teacher's games page. |
| Exceptions: | **User Action** | **System Action** |
| 1. 1- if the user click cancel on the confirmation message |  |
|  | 2 – the system doesn't change |
| 1. 3 - If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 4- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Rate Game | |
| Actors: | Student and teacher | |
| Pre-conditions: | Login , Play Game | |
| Post-conditions: | Notify the teacher who make the game with the rate | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. System will display the rating window after playing the game |
| 2 – The user will rate the game (interesting, normal, and boring). |  |
|  | 3 – system will update the rating of the game to the server |
|  | 4 – the system will notify the teacher who make the game with the rate. |
|  |  | 5 – system will return to the game |
| Exceptions: | **User Action** | **System Action** |
|  | 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 2- System ends the current login session and displays the main interface. |
| Includes | Login ,validation | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Comment on game | |
| Actors: | Student and teacher | |
| Pre-conditions: | Login , Play Game | |
| Post-conditions: | Notify the teacher who make the game with the rate | |
| Flow of events: | **User Action** | **System Action** |
|  | 1. System will display the commenting window after playing the game |
| 2 – The user will comment on the game |  |
|  | 3 – System will save this comment. |
|  | 4 – the system will notify the teacher by the new comment. |
|  |  | 5 – system will return to the game |
| Exceptions: | **User Action** | **System Action** |
|  | 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 2- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Create class room | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | Class room will be created | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class room” tab, from the toolbar, |  |
|  | 1. The System will display this teacher's class room page. |
| 3-User click “Create Class Room” button |  |
|  | 4-System prompts a new interface which contains tools to create new room. |
| 5-The user will enter the need information for creating the class room (name, public or private, subjects,) |  |
| 6– user will click “confirm” button |  |
|  | 7-System will create class room and update to the system. |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user didn't click save button |  |
|  |  | 4- the created game will be neglected |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Delete class room | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | Class room will be deleted | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class room” tab, from the toolbar, |  |
|  | 1. The System will display this teacher's class room page. |
| 3-User click “Delete Class Room” button in the right of the class he want to delete |  |
|  | 4-System ask the user to confirm the deletion process in a new window |
| 5-The user clicks on confirm button |  |
|  | 6-System will delete class room and remove it from the list. |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user didn't click confirm button |  |
|  |  | 4- the process will be neglected |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | Add game(s) to the class room | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | Games will be added to the room | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class Rooms” tab, from the toolbar. |  |
|  | 2-System displays a page contains user’s saved classrooms. |
| 3-User click “Add game(s) “ in front of the wanted room |  |
|  | 4-System prompts list with his\her games |
| 5-The user will select the games which he\she want to add to the room |  |
| 6 – click “confirm” button |  |
|  |  | 7-system will add the games to that room |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user click cancel button in the add games message |  |
|  |  | 4- nothing will be added to the room . |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Remove game(s) to the class room | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | Games will be added to the room | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class Rooms” tab, from the toolbar. |  |
|  | 2-System displays a page contains user’s saved classrooms. |
| 3-User click “Remove game(s) “ in front of the wanted room |  |
|  | 4-System prompts list with his\her games |
| 5-The user will select the games which he\she want to remove from the room |  |
| 6 – click “confirm” button |  |
|  |  | 7-system will remove the games from that room |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user click cancel button in the remove games message |  |
|  |  | 4- Nothing will be removed from the room. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 13 | |
| Use Case Name: | Add student(s) to class room | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | student(s) will be added to the room | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class room” tab, from the toolbar. |  |
|  | 2-System will display a page with list with her\his class rooms |
| 1. User will select then click “Add student(s) “in front the wanted room. |  |
|  | 4-System will prompt search area |
| 5- The user will enter student username |  |
| 6 – click “confirm” button |  |
|  |  | 7-system will add the students to that room |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user click cancel button in the add students window |  |
|  |  | 4- nothing will be added to the room . |
|  |  | 5 – if the system didn’t find the required username prompt error message |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 14 | |
| Use Case Name: | Remove student(s) to class room | |
| Actors: | Teacher | |
| Pre-conditions: | User is logged in as a Teacher | |
| Post-conditions: | student(s) will be added to the room | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class room” tab, from the toolbar. |  |
|  | 2-System will display a page with list with her\his class rooms |
| 1. User will select then click “Remove student(s) “in front the wanted room. |  |
|  | 4-System will prompt search area |
| 5- The user will enter student username |  |
| 6 – click “confirm” button |  |
|  |  | 7-system will Remove the students to that room |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3- the user click cancel button in the Remove students window |  |
|  |  | 4- Nothing will be removed to the room. |
|  |  | 5 – if the system didn’t find the required username prompt error message |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 15 | |
| Use Case Name: | Add friends | |
| Actors: | User | |
| Pre-conditions: | User view the other user profile | |
| Post-conditions: | game will be added to user favorite list | |
| Flow of events: | **User Action** | **System Action** |
| 1- User will click “Add friend” button. |  |
|  | 2-System will add this user to current friends list. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 2- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 16 | |
| Use Case Name: | remove friends | |
| Actors: | User | |
| Pre-conditions: | User view the other user profile | |
| Post-conditions: | will be remove the user from the favorite list | |
| Flow of events: | **User Action** | **System Action** |
| 1- User will click “Remove friend” button. |  |
|  | 2-System will remove this user from current friends list. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 2- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 17 | |
| Use Case Name: | Open class chat room | |
| Actors: | User | |
| Pre-conditions: | User opened class room | |
| Post-conditions: | game will be added to user favorite list | |
| Flow of events: | **User Action** | **System Action** |
| 1- User will click “open chat” button . |  |
|  | 2-System will prompt window that have chat and users in this room. |
| Exceptions: | **User Action** | **System Action** |
|  | 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 2- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 18 | |
| Use Case Name: | View Class Room | |
| Actors: | User | |
| Pre-conditions: | Login | |
| Post-conditions: | Selected Class Room will be displayed | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Class Rooms” tab, from the toolbar. |  |
|  | 2-System displays a page contains user’s saved classrooms. |
|  | 3-If user has a saved class room or more, he clicks on the class room he wants to view. |  |
|  |  | 4- System opens a new page with the contents of the selected class room. |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  |  | 2- System ends the current login session and displays the main interface. |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 19 | |
| Use Case Name: | Search | |
| Actors: | User | |
| Pre-conditions: | Login | |
| Post-conditions: | Games will be added to the room | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click “Search” Textbox, from the toolbar, then enter the text he wants to search for , and click enter |  |
|  | 2-System prompt a page with the matching requesting games |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 3-if system didn’t find related games for the request |  |
|  |  | 4- system will prompt a page will message “didn’t find related word” |
| Includes: | Login | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 20 | |
| Use Case Name: | Compete Friend | |
| Actors: | Teacher , Student | |
| Pre-conditions: | Login , | |
| Post-conditions: | View Score of the game | |
| Flow of events: | **User Action** | **System Action** |
| 1- user Select play game tab |  |
|  | 2- System will display the Subjects list. |
| 3 – user will choose the subjects |  |
|  | 4- system will display the games  Categories |
| 5- user will choose the category |  |
|  | 6 – system will display the games page |
|  | 7-user clicks on compete friend button. |  |
|  |  | 8- the system will display this user friends list in a drop list. |
|  | 9-user will choose the friend he wants to compete with. |  |
|  |  | 10- the system will send a request to this friend |
|  | 11- if this friend will accept this request |  |
|  |  | 12- the system will display a game chosen randomly |
|  | 13- Two users will play the game. |  |
|  |  | 14- the system will prompt result message and the score , increase the total score of the players , show the rate and comment page |
|  |  | 15 – the system prompt two choices  Play again , back to all games |
| Exceptions: | **User Action** | **System Action** |
| 1-If user wants to logout, he clicks on logout button, which exist on the tool bar. |  |
|  | 2- System ends the current login session and displays the main interface. |
|  | 1. If the user's friend refuse the request |  |
|  |  | 1. The system notify the user |
| Includes: | Login , play game | |
| Notes and Issues: |  | |

# Ownership Report :

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Functional Requirements | Nour |
| Use case Model | Mazen |
| Non – Functional | Mahmoud |
| Use case Table | Mohamed |

**The github link :**

https://github.com/supernour09/CleverZone