

**Cairo University  
Faculty of Computers and Information**

**CS251**

**Software Engineering I**

Testy brain

SRS Document

Roqaia Ahmed Abd-Allah

Hagar Mohamed Mohamed

Doaa Ghaleb Noman

Nov 2016

**Team**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140351 | Roqaia Ahmed Abd-Allah | [Roqahmed55@yahoo.com](file:///C:\Users\Therock\Downloads\HYPERLINK%20%22mailto:Roqahmed55@yahoo.com%22Roqahmed55@yahoo.com) | 01021187800 |
| 20140322 | Doaa Ghaleb Noman | [Doaaghaliep.dg@gmail.com](mailto:Doaaghaliep.dg@gmail.com%22Doaaghaliep.dg@gmail.com) | 01015603953 |
| 20140349 | Hagar Mohamed Mohamed | Hagermohamed20mohamed96@gmail.com | 01119385467 |

**Document Purpose and Audience**

This document describing software which is about educational website that contains a lot of games and describe the functional requirements and non-functional

It’s intended for software project group (designer, developers and manager of the website) and also for the client to specify what will be implemented in the Games software.

This system will be designed to maximize knowledge and skills with different fields, games tools to assist the student for training brain to think about solutions and learn a new concepts to challenge yourself to achieve a high scores.

**Introduction**

**Software Purpose**

-The software will be based on educational games that help students while they are playing a very interesting, simple games to understand a basic

- Games will be related to a specific 4 categories which are science, technology (programming), mathematics, and puzzles.

**Software Scope**

-All what are mentions in the requirements are in the scope, otherwise is outside the scope.

-this system is designed to allow the teacher to manage and communicate with students via website.

-The system a web Application it contains a lot of games that’s for a single-player -strategy game on all platforms.

**Definitions, acronyms, and abbreviations**

**SRS document:**

A shortcut for a software requirements specification which is a comprehensive description of the intended purpose and environment for software under development. The SRS fully describes what the software will do and how it will be expected to perform.

**Response time:**  
The time between click button to request system or web site

**Non functional Requirement :**

Non- functional requirements are "qualities", "quality goals", "quality of service requirements", "constraints" and "non-behavioral requirements"

**Requirements**

**Functional Requirements**

The student should be able to :

create an account

play a new game

resume the game

select the level

exit the game

write a comment

rate the game

show the score that have been saved

The Teacher should be able to :

create an account

play a new game

resume the game

select the level

exit the game

respond the comment

Create a new game

Edit game

Remove game

**Create an account**

User shall be able to create account using name, gender, password and birth date.

**Play a new game**

User shall bechoose any game from one of the 4 categories and start playing it.

**Resume the game**

User starts playing from the point he stopped on it.

**Select the level**

User shall be able to select unblocked level

**Exit the game**

User shall be able to leave the game and come back to the main menu.

**Show the score that have been saved**

The system will save the score and shall be able to show it.

**Rate the game**  
User can rate the game (Interesting, good, or board).

**Write a comment**  
User can write a comment.

**Create a new game**

Teacher uploads the information game into the web Application to add a new game.

**Edit game**

The teacher can make any edit on his/her created game.

**Remove game**

The Teacher can remove his/her created game.

**Respond comment**  
The teacher (owner of the game) can replay on any comment written by any student.

**Non Functional Requirements**

**Performance**

Response time is very small (will not exceed more than 2 seconds request operation will be done within 1 second).

**Usability**

Measure number of click easy to use easy interface and game must run on different platforms.

**Reliability**

System will be down maximum in 5 seconds.

**Security**

It’s wrote by encryption algorithm and site log out alone after 30 minutes without any activity.

**Scalability**

System should be able to support up to 1000 simultaneous game player.

**Maintainability**

Teacher make update to game to improve the system to don't have crash which means it's a crash save.

**Accessibility**

Accessibility can be viewed as the "ability to access"

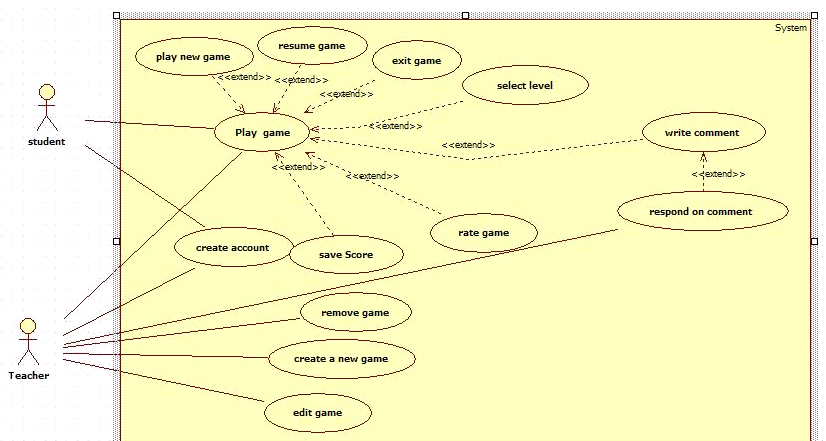
Student and Teacher can access account without any difficult so system is Accessibility.

**Privacy**

System is privacy not allow to any player enter on any account each player enter on privet account only.

**System Models**

**Use Case Model**



**Use Case Tables**

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 1 | |
| Use Case Name: | Create account | |
| Actors: | Student , Teacher | |
| Pre-conditions: | None | |
| Post-conditions: | Student or teacher have an account | |
| Flow of events: | **User Action** | **System Action** |
| 1-User open website |  |
| 2-User press register button |  |
|  | 3- System show register page |
| 4- User enter his/her personal information (first name , last name , password , birth date , gender , E-mail ) |  |
| 5- user should choose if he/she teacher , or student |  |
|  | 6-System should take the code if he/she select the teacher button |
| 7- user should agree the terms of service(Rules and Privacy) |  |
| 8-user press sign up button |  |
|  | 9- System check validity of the information. |
|  | 10- System Send to user confirmation mail |
| 11- user confirm his/her registration by E-mail |  |
|  | 12- System will save account. |
| Exceptions: | **User Action** | **System Action** |
| If user forget any required information |  |
|  | System will respond to fill filed missing information. |
| If user enter incorrect  e-mail |  |
|  | System won’t let user access his/her account till user verified the E-mail . |
| If teacher enter invalid code |  |
|  | System won’t complete registration until teacher enter valid code. |
| Includes: | - | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 2 | |
| Use Case Name: | Create a new game | |
| Actors: | Teacher | |
| Pre-conditions: | Get information of the game | |
| Post-conditions: | Teacher create a game | |
| Flow of events: | **User Action** | **System Action** |
| 1-Teacher enter name and password |  |
|  | 2-System verify teacher's data |
| 3-Teacher select one category |  |
| 4-Teacher uploads the information game into the web Application |  |
|  | 5- System saves the new game |
|  | 6- System add the new game in the selected category. |
|  | **User Action** | **System Action** |
| Teacher enter name and password |  |
|  | Name or password is invalid. |
|  | System reject |
|  | System request to put valid user name or password |
| Teacher forget password |  |
|  | System send a request for user to reset password using the user's email |
| Includes: | Get information game | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 3 | |
| Use Case Name: | Get information a game | |
| Actors: | Teacher | |
| Pre-conditions: | Log in | |
| Post-conditions: | Create new game | |
| Flow of events: | **User Action** | **System Action** |
| 1- Teacher log in |  |
|  | 2- System show profile |
| 3-teacher choose create game |  |
|  | 4-System order information |
| 5-Teacher enter information  (type game, type categories, name of game, age of player can play game |  |
|  | 6-System save information of game |
|  | **User Action** | **System Action** |
| 1-teacher choose type game and age of player |  |
|  | 2-System will show massage to verify age with type |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 4 | |
| Use Case Name: | Edit game | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher create the game | |
| Post-conditions: | Teacher can try the game after he/she edit on it | |
| Flow of events: | **User Action** | **System Action** |
| 1-After the teacher create the game he/she can make any edit on his/her created game. |  |
|  | 2-System save any changes in the game. |
|  | **User Action** | **System Action** |
| Teacher enter name and password |  |
|  | Name or password is invalid. |
|  | System reject |
|  | System request to put valid user name or password |
| Teacher forget password |  |
|  | System send a request for user to reset password using the user's email |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 5 | |
| Use Case Name: | Remove game | |
| Actors: | Teacher | |
| Pre-conditions: | Teacher create the game | |
| Post-conditions: | The game removed from the list of games | |
| Flow of events: | **User Action** | **System Action** |
| 1-After the Teacher created the game he/she can remove it |  |
| 2-Teacher press remove button |  |
|  | 3- System will remove the game from the list of games |
| Exceptions: | **User Action** | **System Action** |
|  | User enter name and password |  |
|  | User name or password is invalid. |
|  | System reject |
|  | System request to put valid user name or password |
| User forget password |  |
|  | System send a request for user to reset password using the user's email |
| If teacher didn’t press remove button |  |
|  | System won’t remove the game |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 6 | |
| Use Case Name: | Play game | |
| Actors: | Student and teacher | |
| Pre-conditions: | User Sign in (have an account) | |
| Post-conditions: | get a score ( rate the game or write a comment) | |
|  | **User Action** | **System Action** |
| 1 - User enter Name and Password |  |
|  | 2- System verify user data |
| 3- User choose any game from one of the 4 categories . |  |
|  | 4- System listed options (new game , resume game , exit the game) |
| 5-User choose one of the options |  |
| 6-User finish the game |  |
|  | 6-System show the achieved score |
| Exceptions: | **User Action** | **System Action** |
| User enter name and password |  |
|  | Name or password is invalid. |
|  | System reject |
|  | System request to put valid user name or password |
| User forget password |  |
|  | System send an E-mail for user to reset password |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 7 | |
| Use Case Name: | Select level | |
| Actors: | Student ,Teacher | |
| Pre-conditions: | User play the game before and the level is unlocked | |
| Post-conditions: | User play within the selected level | |
| Flow of events: | User Action | System Action |
| 1-User go to the main menu of the games |  |
| 2-User choose one from the 4 categories |  |
|  | 3-System will list the games |
| 4-User choose the required game. |  |
|  | 5- System list the options (new game , resume game , select level, exit game) |
| 6- User choose select level option. |  |
| 7- User select level for a game ( easy , normal, hard) |  |
|  | 8-System will display the game dependable on the user selected |
| Exceptions: | User Action | System Action |
| If user didn’t select any level |  |
|  | System by default will make the level easy . |
|  | User can't choose levels that has been unlocked |  |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 8 | |
| Use Case Name: | Save score | |
| Actors: | Student | |
| Pre-conditions: | Student play game | |
| Post-conditions: | Student show the score | |
| Flow of events: | **User Action** | **System Action** |
| 1-after user log in and play a game |  |
|  | 2- System will save his/her score |
|  | 3- System will confirm the user about his/her score |
| Exceptions: | **User Action** | **System Action** |
| If user get out in middle of the game |  |
|  | System won’t save his/her score |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 9 | |
| Use Case Name: | Rate game | |
| Actors: | Student | |
| Pre-conditions: | Student play game | |
| Post-conditions: | System save the rate | |
| Flow of events: | **User Action** | **System Action** |
| 1 - after student play the game  He /she rate it by( interesting , good or bored ) |  |
|  | 2 - system will save the rate. |
| Exceptions: | **User Action** | **System Action** |
| If Student rate but didn’t submit the rate |  |
|  | 2-System won’t save the rate |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 10 | |
| Use Case Name: | Write comment | |
| Actors: | Student | |
| Pre-conditions: | Student play game | |
| Post-conditions: | Teacher respond comment | |
| Flow of events: | **User Action** | **System Action** |
| 1-after user finish game he/she can write a comment |  |
|  | 2- System will mention the teacher who create the game (owner of the game ) to see comment of the student |
| Exceptions: | **User Action** | **System Action** |
| If student didn’t press submit comment |  |
|  | System won’t send the comment to the teacher |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 11 | |
| Use Case Name: | Respond comment | |
| Actors: | Teacher (owner of the game ) | |
| Pre-conditions: | Student write comment | |
| Post-conditions: | Notify the student | |
|  | **User Action** | **System Action** |
|  | System will mention teacher (owner of the game ) |
| Teacher (owner of the game ) will replay on comment |  |
|  | System will notify the student who write the comment |
| Exceptions: | **User Action** | **System Action** |
| If teacher don't respond the comment after specific time (1day) |  |
|  | System notify the teacher again . |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 12 | |
| Use Case Name: | Resume game | |
| Actors: | Student , Teacher | |
| Pre-conditions: | User was played this game before | |
| Post-conditions: | The game stopped | |
| Flow of events: | User Action | System Action |
| 1-User go to the main menu of the games |  |
| 2-User choose one from the 4 categories |  |
|  | 3-System will list the games |
| 4-User choose the required game. |  |
|  | 5- System list the options (new game , resume game , select level , exit game) |
| 6- User select resume game option |  |
|  | 7-System will continue the games from the point he stopped in |
| Exceptions: | User Action | System Action |
| If game cannot loaded and the user waiting for a specific time (2 minutes) |  |
|  | System will get the user out the game . |
| Includes: |  | |
| Notes and Issues: |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | 13 | |
| Use Case Name: | Exit game | |
| Actors: | Student ,Teacher | |
| Pre-conditions: | User play the game | |
| Post-conditions: | User get out of the game and come back to the main menu | |
| Flow of events: | User Action | System Action |
| 1-User go to the main menu of the games |  |
| 2-User choose one from the 4 categories |  |
|  | 3-System will list the games |
| 4-User choose the required game. |  |
|  | 5- System list the options (new game , resume game , select level, exit game) |
| 6- User select exit game option. |  |
|  | 7- System will get user out the game then display the games list |
| Exceptions: | User Action | System Action |
|  |  |
|  |  |  |
| Includes: |  | |
| Notes and Issues: |  | |

**Ownership Report:**

|  |  |
| --- | --- |
| **Item** | **Owners** |
| **Document Purpose and Audience** | Roqaia Ahmed |
| **Software purpose** | Hager Mohamed |
| **Software scope** | Doaa Ghaleb |
| **Functional Requirements/non Functional Requirements** | All team |
| **Use case model and table** | All team |