

**Cairo University**  
**Faculty of Computers and Information**



# **CS251**

# **Software Engineering I**

## **Brain Games**

## **Software Requirements Specifications**

## **Brainiacs**

Mahmoud Mokhtar  
Youssef Mohamed Sanad  
Amr Safwat Sayed Almaz  
Mohanad Magdy Abdelghane  
Omar Salah Ahmed

## **11/2016**



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

# Software Requirements Specifications

## Contents

Team .....	3
Document Purpose and Audience .....	3
Introduction .....	3
Software Purpose .....	3
Software Scope .....	3
Requirements .....	4
Functional Requirements .....	4
Non Functional Requirements .....	6
System Models .....	7
Use Case Model .....	7
Use Case Tables .....	8
Ownership Report .....	22



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

### Team

ID	Name	Email	Mobile
20140320	Mahmoud Mokhtar Hassan	mahmok@gmail.com	01223482084
2014031	Youssef Mohamed Sanad	yosef_sanad@hotmail.com	01096222961
20140184	Omar Salah Ahmed	oziiidan199663@gmail.com	01150182646
20140193	Amr Safwat Sayed Almaz	Amr.almaz2000@gmail.com	01152902450
20140280	Mohanad Magdy Abd-Elghane	mohanadmagdy441@yahoo.com	01113917603

### Document Purpose and Audience

The results of analysis of project are documented. This document describes the functional, non-functional requirements, use cases (events in the software). Also describes the purpose and the scope of the software. This documents acts as an intermediate between the client and developers, so developers can understand the project requirements which the client need. To help developers construct the software effectively and efficiently.

This document is for the client, users, the project management, software developers and software designers.

### Introduction

#### Software Purpose

Playing itself has always been perceived as an activity of enjoyment or recreation rather than a serious or practical purpose, although the activity itself has a potential to be used as a form of experiential learning in a fun and relaxing environment. The main purpose is to represent the idea of learning by playing games to facilitate educational processes. Where children or adults can gain information and learn by playing interactive games which provide them with the knowledge and help them to learn and educate.

#### Software Scope

This software will be a Web Application, served online on the internet so people from all over the world can access this software. This software will include educational games only created by teachers. This software will only gather students and teachers (name, age, gender, study level) and record their information and achievements achieved by students.



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

### Requirements

#### Functional Requirements

##### **1. Students and Teachers should be able to:**

1. Register.
2. Sign in.
3. Search Game.
4. Play Game.
5. Show profile.
6. Edit his\her profile.
7. View games rates.

##### **2. Students should be able to:**

1. Rate Game.
2. Write Comments on games.

##### **3. Teachers should be able to:**

1. Create new games.
2. View his\her games
3. Edit his\her games.
4. Remove his\her games.
5. Respond to his\her game's comments.
6. Delete abusive comments.

##### **1.1 Register:**

Any user can register to the web application through a simple form where he/she can fill with his\her details such as name, age, gender and educational level. Also fills his\her username and password which they will use to sign in later.

##### **1.2 Sign in:**

The user will enter his\her username and password in the sign in form. If they are correct the user will be successfully logged in. If the username and password combination doesn't match the user will be alerted that his\her username or password is incorrect. If the username doesn't exist then application will ask the user to register.

##### **1.3 Search Game:**

After the user is logged in all games will be shown, he\she can press on filter by category buttons, which filter all games and show only the chosen category games. The user can type a game's name in the search box and results will be returned to the user (the game if it is found or an error message if no results found).



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

### **1.4 Play Game:**

The user can click on the game's image, link or title. So he\she can be redirected to the game's page. Each game has instructions, help and menus so the user can start or pause the game chosen.

### **1.5 Show User Profile:**

Users can view his\her profile by pressing on profile tab or button. User must be logged in to do so. Users can view his\her achievements, and other details he\she filled in the registration form.

### **1.6 Edit User Profile:**

Users can edit his\her profile by pressing on the edit button located in the user's profile page. Users can change their Full name, age, gender, password or any other details needed. Updates are saved after the user presses the save button.

### **1.7 View Games Rates:**

Users can see each game's rate while he\she is browsing for games.

### **2.1 Rate Game:**

Students can rate games through a review box\form located in the game's page. The review is submitted after the user presses on submit review button.

### **2.2 Write Comments on Games:**

Students can write comments on games through a comment box\form located in the game's page. The comment is submitted after the user pressed on submit comment button.

### **3.1 Create New Games:**

Teachers can create their own educational games and submit them to the web application.

### **3.2 View Teacher's Games:**

Teachers can view their games by pressing on the "my games" tab or button. Teachers will be redirected to a page listing their created games.

### **3.3 Edit Games:**

Teachers can edit and update their games.

### **3.4 Remove Teacher's game:**

Teachers can remove their created games, which will make students can't play the removed game or find it.



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

### **3.5 Respond to Comments:**

Teachers can respond and reply to students comments on their games.

### **3.6 Delete Rude or Abusive Comments:**

Teachers are able to delete or remove comments containing rude or abusive words only. Any normal comments can't be deleted or removed.

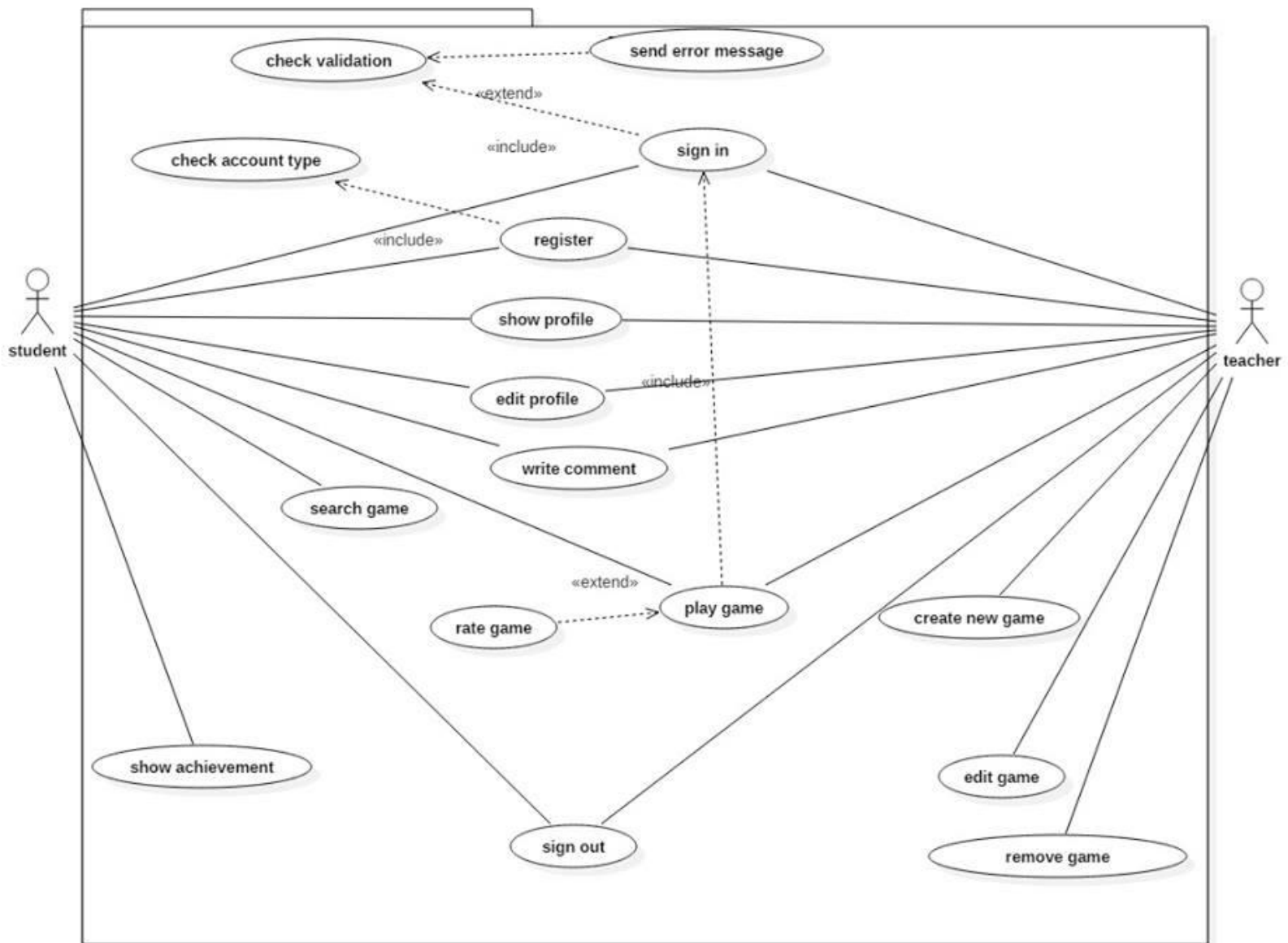
## Non Functional Requirements

1. **Multiple Platform Compatibility:** This web application must be accessed from any device with a browser and internet connection.
2. **Simple to use:** This web application must have simple and friendly user interface to make it easier for users to navigate through the application without any difficulties and also to bring the best user experience.
3. **Performance:** Navigating through the application must be relatively fast, and super responsive.
4. **Crash Safe:** This application must be bug free as much as possible to prevent application crashing.
5. **Documentation:** Software must be well documented. Documentation is really important as it plays a critical role in updating and maintaining this application.
6. **Security:** Databases must be encrypted. Inactivity timer must be set so user will be logged out automatically after 5 minutes of inactivity. During log in if someone enters wrong password and username combination, logging in functionality will be blocked for 3 minutes.
7. **Backups:** This web application will automatically backup user's information weekly.
8. **Extensibility:** This web application can be easily extended and can easily add features.
9. **Pricing:** This web application will be free for users to create an account and play games.
10. **Availability:** This web application will be available 24/7 99% up-time.

# Software Requirements Specifications

# System Models

## Use Case Model





# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

### Use Case Tables

Use Case ID:	1	
Use Case Name:	Sign in	
Actors:	Student,teacher	
Pre-conditions:	<p>The user has an account</p> <p>The user is trying to log in with their account</p>	
Post-conditions:	The user is logged in to the system	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	User accesses the URL	
		The system prompts the user for their account credentials
	The user enters their RIT username and password	
		System check the validation of user name and password
	The user gains access to the systems functionality	
Exceptions:	<b>User Action</b>	<b>System Action</b>
	1- User Enter user name and Password.	
		Incorrect user name or password
Includes:	Check Validation	
Notes and Issues:		





# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	2	
Use Case Name:	Sign out	
Actors:	Student, Teacher	
Pre-conditions:	<p>The user is logged in</p> <p>The user no longer wants to be logged in</p>	
Post-conditions:	The user is logged out	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	The user clicks on the logout button	
		<p>The system logs the user out</p> <p>The system redirects to the default Log out page.</p>
Exceptions:	<b>User Action</b>	<b>System Action</b>
	None	
		None
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	3	
Use Case Name:	Register	
Actors:	Student,teacher	
Pre-conditions:	User is not a member of the website.	
Post-conditions:	User is a member of the website.	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	User navigates to the homepage and selects the 'New User' link.	
		System displays the new user registration screen.
	User enters their email address and password, and the conformation password.	
		System validates that the email does not already exists. If it does not, a new user is created and a confirmation email is sent with a confirmation link that needs to be selected before the user is allowed to enter the site
	User checks their email site for the email that the system sent and gets the confirmation link. The user selects this confirmation link	



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

		System accepts the confirmation link and requests the user to log in using their email and password
Exceptions:	<b>User Action</b>	<b>System Action</b>
	User enters their email address.	
		System validates that the email does not already exists
Includes:	Check Account Type	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	4	
Use Case Name:	Search game	
Actors:	Student	
Pre-conditions:	User is a member in website	
Post-conditions:	User found the game	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	User login	
	User click on search button and write the name of the game	
		The system search for the game and view it in another webpage to start playing
	Start playing game	
Exceptions:	<b>User Action</b>	<b>System Action</b>
	User enter game name	
		System validates that game not found
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	5	
Use Case Name:	Play game	
Actors:	Student,teacher	
Pre-conditions:	User is a member in website	
Post-conditions:	User finish playing	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	Search for game	
		New game is loaded on system
	User click on start bottom and start playing	
Exceptions:	<b>User Action</b>	<b>System Action</b>
	None	
		None
Includes:	Sign in	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	6	
Use Case Name:	Rate game	
Actors:	Student	
Pre-conditions:	User play the game	
Post-conditions:	the game is rated	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	Search for game	
		New game is loaded on system
	User finised play the game then choose the rate of the game	
		System save user rate
Exceptions:	<b>User Action</b>	<b>System Action</b>
	None	
		none
Includes:	Play game	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	7	
Use Case Name:	Write comment	
Actors:	Student, Teacher	
Pre-conditions:	User is a member in website	
Post-conditions:	User send the comment	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	User click on comment button and start writing his comment then click send	
		System upload his comment
Exceptions:	<b>User Action</b>	<b>System Action</b>
	None	
		None
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	8	
Use Case Name:	Show profile	
Actors:	Student , Teacher	
Pre-conditions:	User logged in	
Post-conditions:		
Flow of events:	<b>User Action</b>	<b>System Action</b>
	1- after log in user click on show profile button	
		2- System direct the user to his profile
Exceptions:	<b>User Action</b>	<b>System Action</b>
Includes:	None	
Notes and Issues:		





# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	9	
Use Case Name:	Edit profile	
Actors:	Student , Teacher	
Pre-conditions:	User logged in	
Post-conditions:	Profile is edited	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	1- user clicks on show profile button 2-then click on edit profile button	3-system asks user if he wants to change UserName ,password or profile photo
	4- user Chooses what he wants to change	5- if he choose change password, system asks the user to enter the old password then enter the new password twice
	6- user follow this rules	
Exceptions:	<b>User Action</b>	<b>System Action</b>
	none	
		None
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	10	
Use Case Name:	Show achievements	
Actors:	Student	
Pre-conditions:	User logged in and Played the game	
Post-conditions:	Achievements are showed	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	1- User start playing the game	
		2- game is loaded on the system
	3-show achievement at the end of the game	
Exceptions:	<b>User Action</b>	<b>System Action</b>
	none	
		none
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

### Software Requirements Specifications

Use Case ID:	11	
Use Case Name:	Create new game	
Actors:	Teacher	
Pre-conditions:	User logged in by teacher account	
Post-conditions:	Submit game	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	1- Teacher click on create new game button	
		3- System direct the teacher to a new page to upload the game
	4- browse the game from his computer 5- submit the game	
		6- upload the game on the website 7- add the game to this teacher games
Exceptions:	<b>User Action</b>	<b>System Action</b>
	1- upload wrong file which is not a game	2- Show a message" this is type is not supported "
	3- Error in connection	4- System wait the connection then continue uploading the game
Includes:	None	



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

Notes and Issues:	
-------------------	--

Use Case ID:	12	
Use Case Name:	Edit game	
Actors:	Teacher	
Pre-conditions:	User logged in by teacher account	
Post-conditions:	Remove old version	
Flow of events:	<b>User Action</b>	<b>System Action</b>
	1- Click on “my games” button	
		2- show games
	3- click “edit game” button at the game he wants to edit	4- System direct the teacher to a new page to upload the new version
	5- browse the new version from his computer	7- game is uploaded on the website
	6- submit the game	
	7-	
Exceptions:	<b>User Action</b>	<b>System Action</b>
	none	
		none
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

Use Case ID:	13	
Use Case Name:	Remove game	
Actors:	Teacher	
Pre-conditions:	User logged in by teacher account	
Post-conditions:		
Flow of events:	<b>User Action</b>	<b>System Action</b>
	2- Click on “my games” button	
		2- show games
	3- click “remove game” button at the game he wants to remove	4- request the account password
	5- Enter the password	6- Check the password 7- Remove game
Exceptions:	<b>User Action</b>	<b>System Action</b>
	1- Enter wrong password	
		2- Sign out the account
Includes:	None	
Notes and Issues:		



# CS251: Phase 1 – Brainiacs

## Project: Brain Games

## Software Requirements Specifications

### Ownership Report

Item	Owners
Functional Requirements, Software purpose, Software Scope, non-functional requirements	<i>Mahmoud Mokhtar, Youssef Sanad</i>
Use case model, use case tables	<i>Mohanad, Amr Almaz, Omar Salah</i>