Cairo University  
Faculty of Computers and Information



**CS251**

**Software Engineering I**

CODE BRAIN

Software Requirements Specifications

Alaa Adel Abd El Hafez

Esraa Ragab Abd El Satar

Aya Fawaky Abd El Atty

Esraa Gamal Ragab

11/2016

Contents

[Instructions [To be removed] 3](#_Toc402452668)

[Team 3](#_Toc402452669)

[Document Purpose and Audience 3](#_Toc402452670)

[Introduction 3](#_Toc402452671)

[Software Purpose 3](#_Toc402452672)

[Software Scope 4](#_Toc402452673)

[Definitions, acronyms, and abbreviations 4](#_Toc402452674)

[Requirements 4](#_Toc402452675)

[Functional Requirements 4](#_Toc402452676)

[Non Functional Requirements 4](#_Toc402452677)

[System Models 5](#_Toc402452678)

[Use Case Model 5](#_Toc402452679)

[Use Case Tables 5](#_Toc402452680)

[Ownership Report 6](#_Toc402452681)

[Policy Regarding Plagiarism: 6](#_Toc402452682)

# Team

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Name** | **Email** | **Mobile** |
| 20140065 | Alaa Adel Abd El-Hafiz | Alaa.adel255@gmail.com | 01150249993 |
| 20140052 | Esraa Gamal Ragab | Asraagamal22@yahoo.com | 01222839476 |
| 20140053 | Esraa Ragab Abd El-satar | Esraaragab016@gmail.com | 01285883082 |
| 20140086 | Aya Fawky Abd El-Atty | ayafawky@gmail.com | 01060865439 |

# Document Purpose and Audience

***Purpose:***

This document will describe the **software requirements specifications** (it is more challenging of requirements engineering which requires “the *client* and *users* are expects in their domain and have general idea of what system should do, on the other hand need also *developer* to stratify their general idea”) .then will show **the types of requirements** after specification it .then will use **system models** to simplify the reality throughout this models.

***Audience:***

This document will direct it anyone interested to educational games (as supplier), **client** (actually who supply this project or application), **project manager** to understand each step in this game (from phase specification requirements “which take it from client” until phase of modeling “system models”).

# Introduction

## Software Purpose

The purpose of our Project “Code Brain “ to make Students more exciting about education because our game will encourage them to study (the interface of game attract any person to play a game) so our target from this system to simplify education for all students through games which like it (create another way to education but its loveable from everybody).

## 

## Software Scope

Our scope from our project ( website application) which contains game called “Code Brain” which should provide the simple user interface (easy to use) and this game should works on different operating system, that mean should apply nonfunctional requirements in additional to apply the functional requirements (which consider the base functionality which requested from client to supported in the system).

## Definitions, acronyms, and abbreviations

***Definitions:***

*Website:*

A place on the World Wide Web which consists of many pages and contains or describes specific information about person or organization.

*Game:*

Is an activity that specific rules that people do for pleasure.

*MP:*

First type from 3 types of game in our website.

MCG:

Second type from 3 types of game in our website.

RCG:

Third type from 3 types of game in our website.

***Acronyms and abbreviations:***

*MP:*

Stands for match picture.

*MCG:*

Stands for multiple choice game.

*RCG:*

Stands for Run code game.

# Requirements

## Functional Requirements

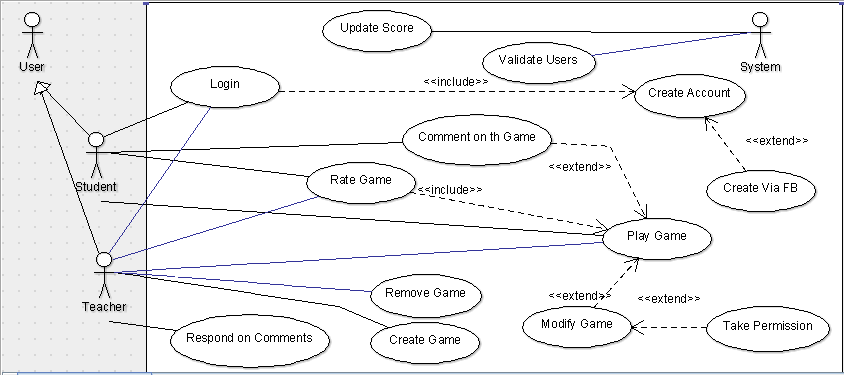
* **Create account:**
* Determine if he/she Student or Teacher.
* He/she mustAdd Personal information’s like Name, Age, and Gender.
* Student or Teacher Have choose to log in with Facebook account or using E-mail.
* **Teacher:**
* Can try any Game inside the Web site.
* Can comment on any other game, and the game owner replay to him.
* Can modify his game and other games upon permission from the Game owners.
* Can rate the game after he/she try it.
* **Create Game:**
* The Teacher can create a game by selecting any programming language he need.
* Then he should create user interface, then implement game.
* After implementing the game he should display it for the web site users.
* Determine the category of the game, The Category Classified in to three parts first is Match pictures, second is Multiple Choices game, and third is Run code game.
* Teachers will add coding games section which help the students To Know the basic programming concepts.
* **Student:**
* play Game by choosing the desired Category (match pictures-multiple choice-run code)
* Can modify game by adding features on its code but first he/she should have permission from the owner of the game “Teacher”.
* Can rate any game from his/her Point of View and he/she can rate using this options (interesting, normal, and boring)
* Student Can Comment on any game and the Teacher should be able to respond to Students’ comment.
* The Student Has option to Write Comment For each game if it’s boring ,interesting , Slow needed Features , can’t be understand , easy ,fast, Hard or any other comment he want.
* **System:**
* The system should contain the Score of The Game that he/she get (student’s achievements).
* Update Student Score and Display it in his profile.
* Summation student points in all games.
* The system will arrange Student according to their score and Display “Top 50” in Teachers and Students Profile.

## Non Functional Requirements

|  |  |
| --- | --- |
| **Reliability** | * **Our website is educational website games so it must be available and credibility to be Flourish.** * **Crash safe. The website application” which include Code Brain game” should be crash safe in 90% of its runtime.** * **There is daily backup to save data regularly.** |
| **Usability** | * **Simple and easy to use.** * **If Teacher or student want to create a game he/she have all tools he need to create or modify the game.** |
| **Security** | * **If the user take long time 15 sec without access the website the system should display time out and log out immediately.** * **We build our system using PHP programming language instead of java language to be more secure.** * **Each account have a unique password and user name to access on web site.** |
| **Legal** | * **Cheap Game. The development of our system “web site” must be done with a budget more than or equal 1200Þ. As the games should reach as many Students as possible, also we should support our education and enhance it, so it can’t be expensive.** |
| **Performance** | * **Should the response time quick for our system (that means the all response times of our website must less than 3 seconds). This will affect on availability) when anybody access the website should be available at any time not failed) and accuracy.** |
| **Reliability &Robustness** | * **The system will be score the student game so It should be credible and have the ability to**   **Maintain the function, say if the user fail to play the score should be zero and vice versa.** |
| **supportability** | * **the Student able to add rate the game (interesting- boring- hard-easy), so inside our web site there will be multiple optional to rate the game and add change on it but changes must be specific like changing color of the game, speed of game, font of the word and we should be considered this options and make limits on it, so the user (teacher or Student) have ability to change but we have limit on it(that’s mean our website support adaptability), students and teachers can modify the game (that’s mean our website support maintainability). So we will get our website has features:** **adaptability (ability to change in our website with additional application domain concepts) and maintainability (ability to develop website with new technology or fix defects).** |
| **Availability** | * **Our website should be available at any time and well performed , because anybody(e.g. :teacher and students) can access the website so the website shouldn’t failed when access from anybody)** |
| **Scalability** | * **The system able to perform under an increased or expanding workload(up to 10000 user).** |

# System Models

## Use Case Model

****

## Use Case Tables

|  |  |  |
| --- | --- | --- |
| Use Case ID: | one | |
| Use Case Name: | login | |
| Actors: | User (Teacher and Student) | |
| Pre-conditions: | should have an account on the website(create account) | |
| Post-conditions: | User can perform any transaction(play games and perform any feature in the website) on the website | |
| Flow of events: | **User Action** | **System Action** |
| 1- User click on button login in the home page of the website |  |
|  | 2- System displays the login page |
| 3- user enter username and password |  |
|  | 4-System validates the login information(username,password) against account table in data base |
|  | 5-Display the personal page for the user |
| Exceptions: | **User Action** | **System Action** |
| 1- User enter invalid username, password |  |
|  | 2- system display invalid message then request from user to re-enter the information again |
| Includes: | two |  |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | Two | |
| Use Case Name: | Create Account | |
| Actors: | User (Teacher and Student) | |
| Pre-conditions: | Display Website Page | |
| Post-conditions: | Choose if teacher or student  Choose create new account | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enter name , password, email |  |
|  | 2- System send verification mail |
| 3- user enter verification code |  |
|  | 4-System verify code |
|  | 5-Display Done message |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter invalid username , password or email |  |
|  | 2- system display invalid message |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | Three | |
| Use Case Name: | Play game | |
| Actors: | User (teacher or student) | |
| Pre-conditions: | Login | |
| Post-conditions: | - Rate game  - Modify game  - add comment | |
| Flow of events: | **User Action** | **System Action** |
| 1- User login as teacher or student |  |
|  | 2- System Verify user data |
| 3- Choose game category. | 4-show list of games in this category |
| 5-Select Game. | 6-System Display game. |
|
| Exceptions: | 1-if user enter invalid information in form of login |  |
|  | 5-The system displays wrong message then request enter the information again |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use Case ID: | Four | | | |
| Use Case Name: | Rate game | | | |
| Actors: | User (Teacher and Student) | | | |
| Pre-conditions: | Must play the game first | | | |
| Post-conditions: | Can add comment | | | |
| Flow of events: | **User Action** | | **System Action** | |
| 1- User plays the game | |  | |
|  | | 2- System displays the rate categories | |
| 3- user select the specific category of the rate (press button ok) | |  | |
|  | | 4-system displays message the rate is done | |
| **User Action** | | **System Action** | |
| Exceptions: | 1-user select specific category of rate (press ok) but this rate not saved if user has problems in connection | | 2-system display wrong message (rate not done …..try again ) | |
| **Use Case ID:** | **Five** | | | |
| Use Case Name: | Create game | | | |
| Actors: | Teacher only | | | |
| Pre-conditions: | Must have Account and be a teacher | | | |
| Post-conditions: | * Replay to comments on it * Modify the game * Try game | | | |
| Flow of events: | **User Action** | | **System Action** | |
| 1-user select create game. | |  | |
|  | | 2-system displays the creation form | |
|  | 3- user select programming language, category of the game, create user interface | |  | |
|  |  | | 4-system displays message (the creation of game is done) | |
|  |  | | 5-system put the new game on the website | |
| Use Case ID: | | six | | | |
| Use Case Name: | | Update score | | | |
| Actors: | | System | | | |
| Pre-conditions: | | After the user playing the game | | | |
| Post-conditions: | | - Sum student points in all games and update it to the student profile.  - Display top 50 | | | |
| Flow of events: | | **User Action** | | **System Action** | |
| 1-The system should contain the Score of The Game that he/she get (student’s achievements). | |  | |
| 2- Sum student points in all games. | |  | |
| 3- Update Student Score and Display it in his profile. | |  | |
| 4- The system will arrange Student according to their score and Display “Top 50” in Teachers and Students Profile. | |  | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | seven | |
| Use Case Name: | Remove game | |
| Actors: | Teacher only | |
| Pre-conditions: | should have an account on the website(create account) | |
| Post-conditions: | The game disappeared from the website | |
| Flow of events: | **User Action** | **System Action** |
| 1- teacher select the remove option for the delete the specific game |  |
|  | 2- System will remove game from website |
| Exceptions: | **User Action** | **System Action** |
| 1- teacher enter name of removed game |  |
|  | 2- system display wrong message the game already removed |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | eight | |
| Use Case Name: | Respond on comments | |
| Actors: | Teacher only | |
| Pre-conditions: | The notification will send to teacher about comments | |
| Post-conditions: | The Replies will send into student will comment | |
| Flow of events: | **User Action** | **System Action** |
| 1-student comment into any game |  |
|  | 2- System will prepare notification into teacher who create this game |
| 3- teacher replay into comments |  |
|  | 4-System send notification the students which write comments |
| Exceptions: | **User Action** | **System Action** |
| 1-student comment into any game but this comment not submit because any problems in connection |  |
|  | 2- system display message fail to comment |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | Nine | |
| Use Case Name: | modify game | |
| Actors: | User(Teacher and student) | |
| Pre-conditions: | User should play the game | |
| Post-conditions: | Game has a new modification | |
| Flow of events: | **User Action** | **System Action** |
| 1- user play the game |  |
|  | 2- System displays option of modify game |
| 3- user select modify game |  |
|  | 4-System will send request into teacher who create this game to take permission |
|  | 5-after take permission the system will display message(the successful modification) |
| Exceptions: | **User Action** | **System Action** |
| 1-user select modify game |  |
|  | 2- System will send request into teacher who create this game to take permission but it don’t take permission. |
|  |  | 3-the system display message (the modification is failed) |
| Extends: | Take permission (Thirteenth) |  |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | Ten | |
| Use Case Name: | Validate users | |
| Actors: | System | |
| Pre-conditions: | After the user login | |
| Post-conditions: | - give each user his powers. | |
| Flow of events: | **User Action** | **System Action** |
| -after the user login the system validates his data to make sure if he is student or teacher, because the teacher only have the authority to add or remove game |  |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | Eleven | |
| Use Case Name: | Create via Facebook | |
| Actors: | User (Teacher and Student) | |
| Pre-conditions: | Display Website Page, have Facebook account | |
| Post-conditions: | Choose if teacher or student | |
| Flow of events: | **User Action** | **System Action** |
| 1- User connect the website with Facebook |  |
|  | 2- System send verification mail |
| 3- user enter verification code |  |
|  | 4-System verify code |
|  | 5-Display Done message |
| Extend | - one | |

|  |  |  |
| --- | --- | --- |
| Use Case ID: | Twelve | |
| Use Case Name: | Comment on game | |
| Actors: | User (Student) | |
| Pre-conditions: | Open game | |
| Post-conditions: | Responded by the teacher | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enter name , password, email |  |
|  | 2- System send verification mail |
| 3- user enter verification code |  |
|  | 4-System verify code |
| 6-open game(may play game or not) and comment on the game | 5-Display Done message  7-system send notification to the teacher that a student commented on a game(the teacher may respond to the comment) |
| Exceptions: | **User Action** | **System Action** |
| 1- User Enter invalid username , password or email |  |
|  | 2- system display invalid message |
| Extend | Play game | |

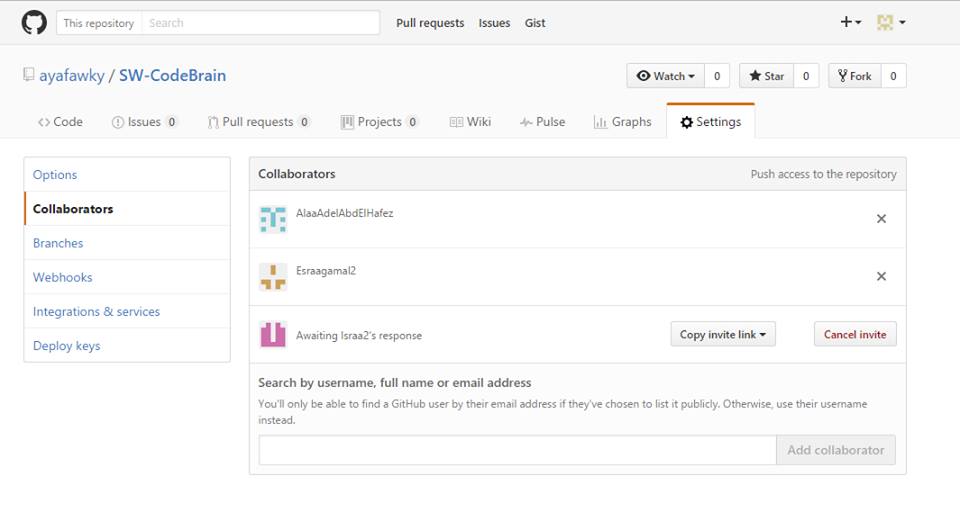
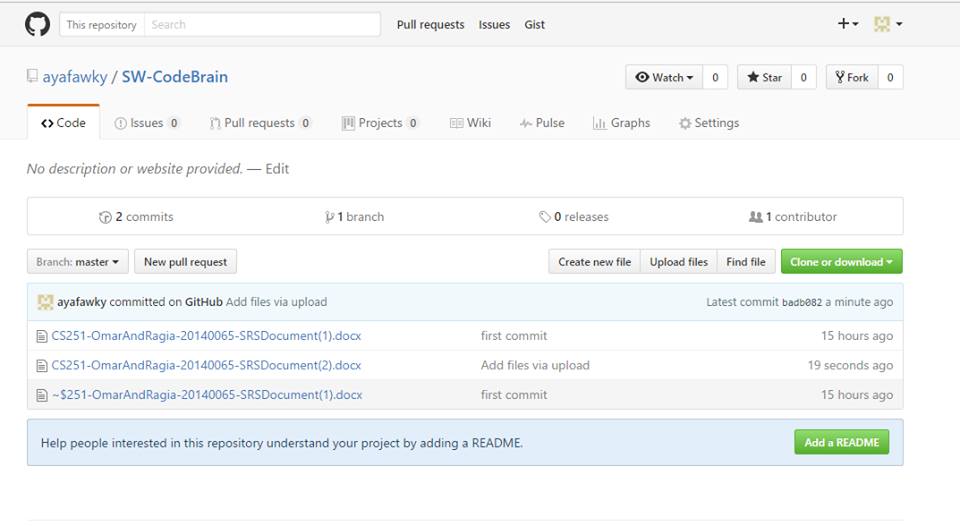
|  |  |  |
| --- | --- | --- |
| Use Case ID: | Thirteenth | |
| Use Case Name: | Take permission | |
| Actors: | User (Student or teacher) | |
| Pre-conditions: | Open game | |
| Post-conditions: | Permission accepted or not | |
| Flow of events: | **User Action** | **System Action** |
| 1- User enter name , password, email |  |
|  | 2- System send verification mail |
| 3- user enter verification code |  |
|  | 4-System verify code |
| 6-open game, play game, choose to modify game and he isn’t the owner of the game  8-the owner of the game may accept the request or not | 5-Display Done message  7-system send notification to the teacher that created the game |
| Extend | Modify game |  |

* Ownership Report

|  |  |
| --- | --- |
| **Item** | **Owners** |
| Document Purpose and Audience  Introduction | *Esraa Gamal* |
| Requirements | *Esraa Ragab* |
| Use Case Model | *Aya Fawky* |
| Use Case Tables | *Alaa Adel* |

# Policy Regarding Plagiarism:

1. ساعد زملاءك على قدر ما تستطيع و حل لهم مشاكلهم فى الكود و لكن تبادل الحلول غير مقبول و يعتبر غشا.
2. أى حل يتشابه مع أى حل آخر بدرجة تقطع بأنهما منقولان من نفس المصدر سيعتبر أن صاحبيهما قد قاما بالغش.
3. قد توجد على النت برامج مشابهة لما نكتبه هنا أى نسخ من على النت يعتبر غشا يحاسب عليه صاحبه.
4. إذا لم تكن متأكدا أن فعلا ما يعد غشا فلتسأل المعيد أو أستاذ المادة.
5. فى حالة ثبوت الغش سيأخذ الطالب سالب درجة المسألة ، و فى حالة تكرار الغش سيرسب الطالب فى المقرر.

https://github.com/ayafawky/SW-CodeBrain