Cairo University

Faculty of Computers and Information

CS251

Software Engineering I

EduGaming

November2016

CS251: Phase 1 – CodeCraft
Project: <EduGaming>

GitHub Link

https://github.com/islam-Ellithy/EduGaming

Team

ID	Name	Email	Mobile
20140116	Rami Mohamed	ramimohamedsalah96@gmail.com	01149273153
20140109	Khaledyousef	khaledyousef196@gmail.com	01114300109
20140056	Islam Emam	islam.emmo@gmail.com	01127321234
20140128	RemonAtef	remonatef128@yahoo.com	01285129556

Document Purpose and Audience

Project declaration and Showing a dynamic design for how the system should work.

Audience :Computech Project Manager.

Introduction

Software Purpose

Our purpose is to develop easy web based software that can change the boring teaching techniques into fun learning one by developing a software that can create a link between teachers and students through building simple games which doesn't need past experience to be built only by dragging and dropping then they can upload it for students to play it and learn their subjects through it .

Software Scope

A teacher–student website that can teach students two different fields programming and science, By developing some simple tools that can be easily dragged and dropped for example entities and functions that can be used for building certain challenging games for students or by writing simple codes with missing parts and supported by choices to give students chance to try coding and learn from their mistakes as for programming field.

Definitions, acronyms, and abbreviations

Definition	Description
Fps	Frame per second

Requirements

Functional Requirements

Website must categorize user type to check if user is teacher or student.

-If teacher then he/she can choose the category of programming or science then create the desirable game by dragging and dropping some tools.

-Teacher can save the created game or exit without saving.

-System must respond to teacher that game has been saved correctly when he tries saving.

-If student then he/she can choose the category of programming or science thenchoose the game then solve the created game by teacher.

-Any user can select the category of games.

-Guest can try any game for a limited time.

Non Functional Requirements

-System should Authenticate and validate user's identity as it's mandatory for the user to enter authentication key in login.[reliability]

-System Should differentiate between Student and Teacher in Interface (Teacher has the ability in his/her interface to create games).[usability]
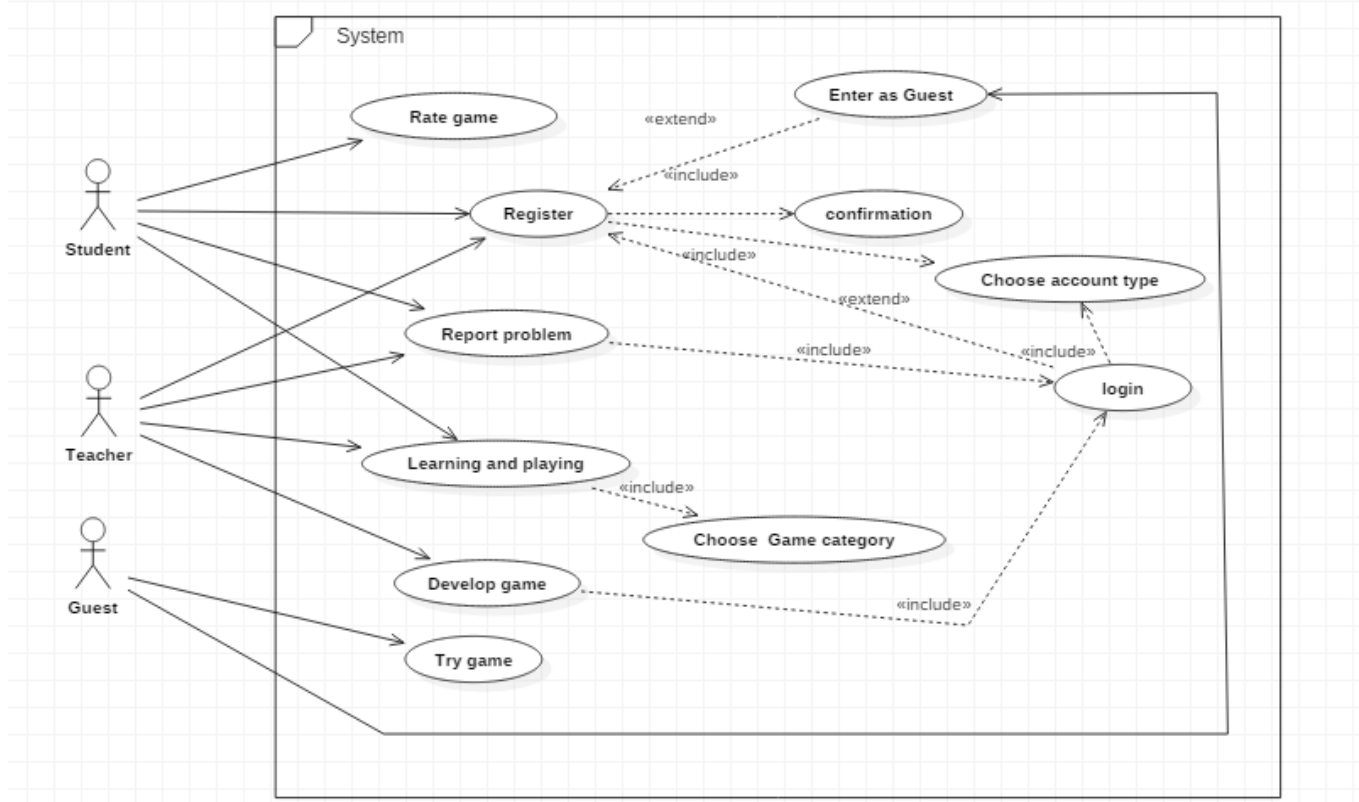
-Saved games cannot be removed from the system when the system goes down.[reliability]

-When teacher tries to leave without saving system must double check his desire.[reliability]

-Server must be stable and available for use any time.[reliability]

-User's ability to report for website problems.[supportability]

-System's response cannot exceed 10 seconds.[performance]

-System loads games with rate higher than 30 fps.[performance]

System Models

Use Case Model

Use Case Table

Use Case ID:	1	
Use Case Name:	Registration	
Actors:	Student ,Teacher	
Pre-conditions:	Visit website	
Post-conditions:	Valid registration confirmed.	
Flow of events:	**User Action**	**System Action**
	1- Make register request	
		2- System loads registration type selection page
	3-Choose registration type	
		4-System loads registration page
	5- Enter personal information and authentication	
		6- System check for validation and redirect to login page
Exceptions:	**User Action**	**System Action**
	1- Enter personal information and authentication [invalid or used information]	
		2- Highlight invalid data in registration. (e.g. username invalid)
	1- Enter as a guest.	
		2- System postpones registration.
Includes:	Confirmation.	

Use Case ID:	2	
Use Case Name:	Learning and playing	
Actors:	Student,Teacher	
Pre-conditions:	Check for registration	
Post-conditions:	Game stopped	
+Flow of events:	**User Action**	**System Action**
	1- Login	
		2- System Verifies for login information and redirects to student page.
	3- Choose desired category.	
		4- System shows games within selected category.
	5- Choose desired game.	
		6- System starts game.
	7-Play game 8-Exit game	
		9-System redirect to student page.
Exceptions:	**User Action**	**System Action**
	1- Complete game	
		2- System adds achievement.
	1-Pause game	
		2-System wait.

Use Case ID:	3	
Use Case Name:	Build game	
Actors:	Teacher	
Pre-conditions:	Visit website, Check for registration	
Post-conditions:	Save game contents	
Flow of events:	**User Action**	**System Action**
	1-login.	
		2- System Verifies for login information and redirects to teacher page.
	3-Select game creation page	
		4-System opens the creation page
	5-Select category of the desired game.	
		6-System shows the game categories' tools (entities and functions)
	7-Build desired game (Drag and drop entities and functions).	
	8-Select create built game.	
		5-System adds the game to the student page and redirects to teacher page.
Exceptions:	**User Action**	**System Action**
	1- Close website before saving game.	

		2- System pops up a warning message assuring user's intent to close website without saving.

Use Case ID:	4
Use Case Name:	Rate game
Actors:	Student
Pre-conditions:	Has played the game
Post-conditions:	The game has been rated from the student

Flow of events:	**User Action**	**System Action**
	1- Push rate game button	
		2- System displays form of rating
	3-Select the game rate	
		4-Systemsaves the rating for the game for that user

Use Case ID:	5
Use Case Name:	Report a problem
Actors:	Student ,Teacher
Pre-conditions:	Login.
Post-conditions:	Problem solved.

Flow of events:	**User Action**	**System Action**

	1- Make report ticket.	
		2- System provides report type choices.
	3-Select report type 4-Write description about the problem and submit it.	
		4-System shows message thanking user for submitting report and sends the report for the technicians.

Use Case ID:	6	
Use Case Name:	Try Game	
Actors:	Guest	
Pre-conditions:	Visit website	
Post-conditions:	Game stopped, Trial time end	
+Flow of events:	**User Action**	**System Action**
	1- Enter website as a guest	
		2- System redirects to category page.
	3- Choose desired category.	
		4- System shows games within selected category.
	5- Choose desired game.	
		6- System starts game and counts down trial time.

	7-Play game 8-Exit game or trial time ends	
		9-System pops up message telling user to register if he/she liked website.
Exceptions:	**User Action**	**System Action**
	1-Pause game	
		2-System wait.

Ownership Report

Item	Owners
Document Purpose and Audience	*RemonAtef*
Software Purpose	*KhaledYousef*
Software Scope	*KhaledYousef*
Definitions, acronyms, and abbreviations	*Rami Mohamed*
Functional Requirements	*RemonAtef*
Nonfunctional Requirement	*All team members*
Use case Model	*All team members*
Use case 1	*Khaledyousef*
Use case 2	*Rami Mohamed*
Use case 3	*Islam Emam*
Use case 4	*Islam Emam*

Use case 5	*RemonAtef, Islam Emam*
Use case 6	*Rami Mohamed*