Cairo University

Faculty of Computers and Information

CS251

Software Engineering I

Software Requirements Specifications

Amira Sherif Alsayed

Dina Essam Eldin

Youmna Essam Fawzy

Nouran Mohssen el-Halawanny

November, 2016.

CS251: Phase 1 – Eden
Project: Bright Minds

Software Requirements Specifications

Contents

CS251: Phase 1 – Eden
Project: Bright Minds

Software Requirements Specifications

Team

ID	Name	Email	Mobile
20140076	Amira Sherif Alsayed	amiiraelsayedd@gmail.com	01271309802
20140333	Dina Essam Eldin	dododandoqa@gmail.com	01114180062
20140315	Youmna Essam Fawzy	yomna_essam80@yahoo.com	01022758669
20140294	Nouran Mohssen el-Halawanny	nouran_elhalawany@hotmail.com	01141515043

Document Purpose and Audience

- **Document purpose**: This document is the Initial Software Specifications Document, it captures complete description about how the system is expected to perform. This software system is a game-based educational platform. The document shows the software purpose, scope, functional and non-functional requirements which allow users to play a game for its educational purpose. It also shows the use case model and the use case tables that describe the actions of the software.

- **Audience**: CEO. Dr Amr Kamel, The clients TA/ Nora Abd El-Hameed and TA/ Desoky Abd El-Qawy.
  The software developers, the software analysts, sponsors and tester.

CS251: Phase 1 – Eden
Project: Bright Minds

Software Requirements Specifications

Introduction

Software Purpose

- The software is a non-profit game based educational platform, developed with the aim of providing skills and knowledge of a certain subject to users by solving interactive exercises in the form of playing a game that support the basics of that subject. The software also allows users if they are teacher to add new games to the website and enhance their old ones.

Software Scope

- **Users:**
    - Student.
    - Teacher.
- **Sign In:**
    - Email / Account name.
    - Password.
- **Sign Up:**
    - Name – Gender – Phone – Age – Nationality.
    - Email.
    - Password.
    - Student's school degree / Teacher's serial number.
- **Categories:**
    - Math.
    - Science.
    - Coding.
    - Technology.
- **Games Types:**
    - Match pictures.
    - Multiple Choices game.
    - Run code game.
- **Games Options:**
    - For Students: Play – Comment – Rate – Display score.
    - For Teachers: Play – Edit – Remove – Respond to comments.

Software Requirements Specifications

Definitions, acronyms, and abbreviations

Accounts	Users should register in system to be able to play (for student) and add games (for teachers).
Sign In	Users who already have an account created in the system can get access to.
Sign Up	Users who don't have an account created in the system can create a new one to get access.
Categories	The subject/type of games that the user should choose between.
Complexity	The user can choose the level of complexity of each game whether it easy, intermediate or advanced.
Rate	The user can say his opinion by rating a game and saying whether it's boring, normal or interesting.
Score	The points the user earned by playing the game and completing levels in.
Database	All the information of the users saved on the system.
Quality	The resolution of the games graphics.
Graphical Interface	A type of user interface that allows users to interact with the system through graphical icons and visual indicators.

Software Requirements Specifications

Requirements

Functional Requirements

- Website asks user (Teacher or Student). → (1)
- User chooses his type. → (2)
- User will fill the signing in information. → (2)
- While signing in the system will access the database to check if the info is right. → (5)
- User will sign up if he hasn't account and fill the required information (Name, Gender, Phone, age, mail, Nationality, School degree if the user is student, serial number if the user is teacher). → (2)
- While signing up the system will save the new account information to the database, check if the info already exists and check the serial number if the user is teacher. →(5)
- User will choose the Category. → (2)
- System will list all the available games in this category. → (3)
- If the user is a Teacher:
    - He can create new game and determine optional complexity levels to his game. →(2)
    - He can try any game. → (3)
    - He can Edit / Remove his games. → (4)
    - He can respond to students' comments on his games. → (2)
- If the user is a Student:
    - He can play any game. → (3)
    - He can rate / comment on any game. → (2)
    - He can choose the complexity level in each game before playing. →(3)
- The website will provide the user with optional qualities in each game to choose from them. → (5)
- After playing a game the system will save the student's score. →(3)
- The system will show the student's score after playing any game. →(3)
- After playing a game the system will show the Top 10 players' scores (if the student requested). →(3)

Software Requirements Specifications

Non Functional Requirements

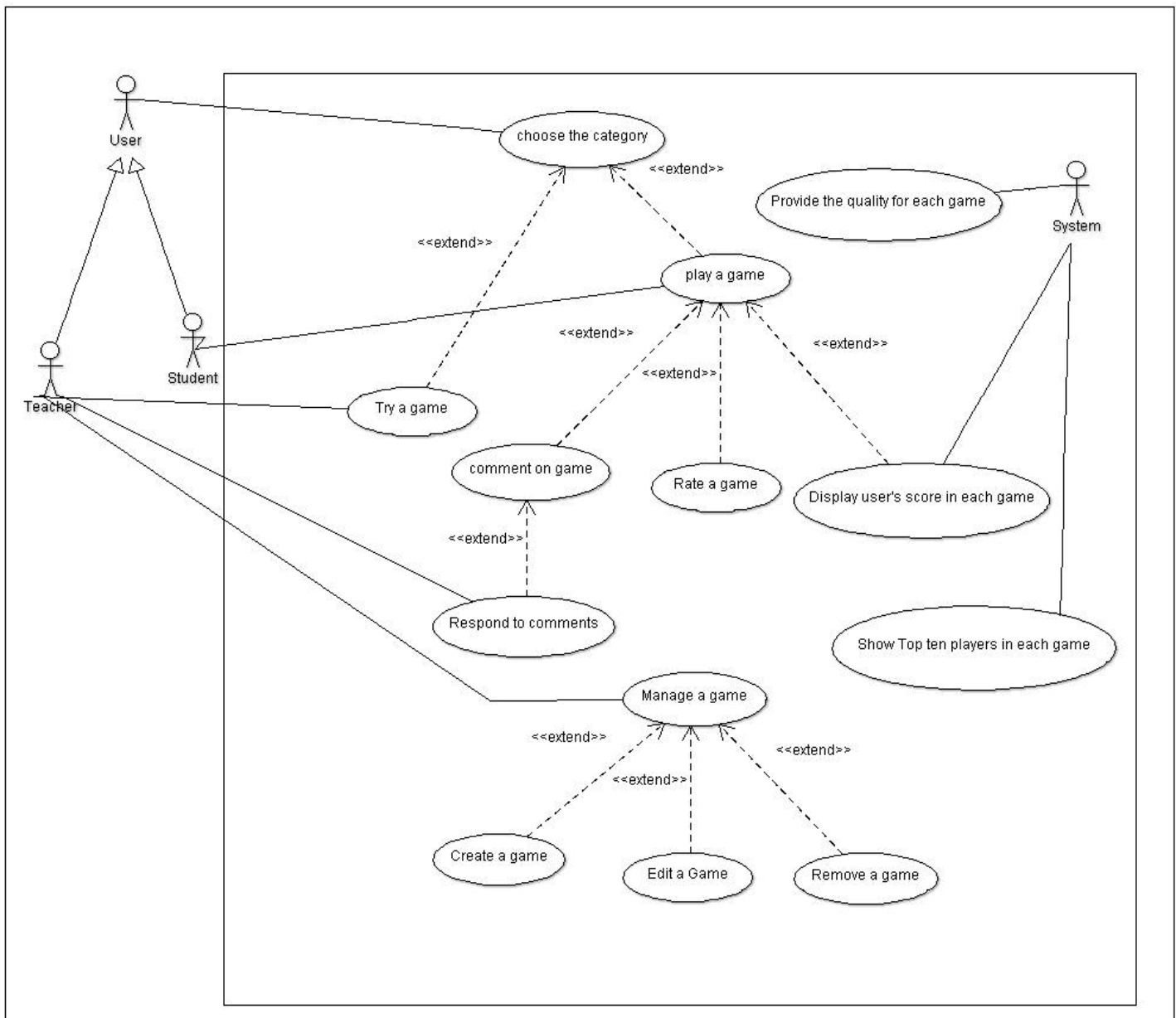|	Details
Performance	- Any Page Will Load within 0.05 to 5 sec. - The system accesses database within 2 to 10 sec. - The game take 1 min to be uploaded on the system by the teacher- - when removing game from the website it will take 10 to 30 sec. - - Any Game will load within 10 sec to 2 min. - Time to show the high score of the student in specific game will take 1 sec or less. - Time to show the Top 10 players' scores on a specific game will take 1 sec or less. - The website will change from one quality to another within 10 sec to 2 min. - The system takes within 1 to 2 sec to upload any comment, rate, respond in any game in the website.
Scalability	- The system allow 1000 user to play any game at the same time. - The system allows 1000 user to rate, respond and comment in any game at the same time.
Security	- The system should protect the users data stored in their profiles by only allow the authorized person who knew the profile password to login. - When a teacher is signing up the system will ask for the teachers serial number , check it and verify the signing up process
Usability	The system must have a graphical interface that the users can easily use and do whatever he want without facing many problems in how to do it.

Software Requirements Specifications

System Models

Use Case Model

Software Requirements Specifications

Use Case Tables

Use Case ID:	1	
Use Case Name:	Choose category	
Actors:	User	
Pre-conditions:	User should has an account and open it.	
Post-conditions:	Choose correct category	
Flow of events:	**User Action**	**System Action**
	1- User Enter User name and Password.	
		2- System Verify user data
	3-User Enter his choice "Teacher" or "Student".	
	4- User Select category of the game he/she want.	
		5-System displays all games in chosen category.
Exceptions:	**User Action**	**System Action**
	1- If User choose that he is Student.	
		2-System allow the features of Student (play, rate or comment on a game).
	3- If User choose that he is Teacher.	
		4-System allow the feature of Teacher (manage a game, try or response to the comments).

Software Requirements Specifications

Use Case ID:	2	
Use Case Name:	Try a game	
Actors:	Teacher	
Pre-conditions:	Teacher should sign in and choose the category of the game.	
Flow of events:	**User Action**	**System Action**
	1- Teacher Enter user name and Password.	
		2- System Verify teacher data
	3- Choose Teacher type	
		4- System open the options to choose category and try any game.
	5-Teacher choose the game he/she want to try.	
Includes:	Extend from Use case 1.	

Software Requirements Specifications

Use Case ID:	3	
Use Case Name:	Play a game	
Actors:	Student	
Pre-conditions:	Student should have an account.	
Flow of events:	**User Action**	**System Action**
	1- Student Enter User name and Password.	
		2- System Verify student data.
	3- User Select his type "Student" and choose category of the games.	
		4- System displays all games of chosen category.
	5-Student can choose the game and its levels and play it.	
Includes:	Extend from Use case 1.	

Software Requirements Specifications

Use Case ID:	4	
Use Case Name:	Displays user's score.	
Actors:	System	
Pre-conditions:	Student ask about his/her score.	
Flow of events:	**User Action**	**System Action**
	1- Students play games.	
		2- System save and list the scores.
		3- System retrieves the score of student.
Exceptions:	**User Action**	**System Action**
	1- Student never play a game.	
		2- System refused to retrieve the score.
Includes:	Include from Use case 3.	

Software Requirements Specifications

Use Case ID:	5	
Use Case Name:	Rate a game	
Actors:	Student	
Pre-conditions:	Student play this game.	
Flow of events:	**User Action**	**System Action**
	1- Student play a game.	
		2- System displays an option to rate or comment on the game.
	3- Student choose to rate on a game.	
		4- System save his rate.
Exceptions:	**User Action**	**System Action**
	1- Student rate the game he/she never play.	
		2- System refused to allow student to rate.
Includes:	Extend from Use case 3.	

Software Requirements Specifications

Use Case ID:	6	
Use Case Name:	Comment on a game	
Actors:	Student	
Pre-conditions:	Student play this game.	
Flow of events:	**User Action**	**System Action**
	1- Student play a game.	
		2- System display an option to rate or comment on the game.
	3- Student choose to comment on a game.	
		4- System save his comment and allow teacher to respond to it.
Includes:	Extend from Use case 3.	

Software Requirements Specifications

Use Case ID:	7	
Use Case Name:	Responds to comments.	
Actors:	Teacher	
Pre-conditions:	Student write comments.	
Flow of events:	**User Action**	**System Action**
	1- Student play and write a comment on the game.	
		2- System save the comment and display it to teacher.
	3- Teacher responds to these comments.	
		4- System displays the responds to students.
Includes:	Extend from Use case 6.	

Software Requirements Specifications

Use Case ID:	8	
Use Case Name:	Manage a game	
Actors:	Teacher	
Flow of events:	**User Action**	**System Action**
	1- Teacher should sign in and choose the "Teacher" type and choose category.	
		2- System displays all features that teacher can do.
	3- Teacher can choose to create, edit or remove a game.	

Software Requirements Specifications

Use Case ID:	9	
Use Case Name:	Edit a game	
Actors:	Teacher	
Pre-conditions:	The game is already exist.	
Post-conditions:	The edition not change the category that the game belongs.	
Flow of events:	**User Action**	**System Action**
	1- Teacher should sign in and his type and choose category.	
		2- System displays all the games in the chosen category.
	3- Teacher choose a game and edit on it.	
		4- System allow the teacher to edit.
Exceptions:	**User Action**	**System Action**
	1- Teacher make changes that change the type of this game.	
		2- System display error message to change the category-if exist- or to discard the changes.
Includes:	Extend from Use case 8.	

Software Requirements Specifications

Use Case ID:	10	
Use Case Name:	Create a game	
Actors:	Teacher	
Pre-conditions:	Teacher should sign in.	
Flow of events:	**User Action**	**System Action**
	1- Teacher Enter User name and Password.	
		2- System Verify teacher data.
	3- Teacher choose to create a game.	
	4-Teacher can add feature of the levels of the game and try to play the game after the creation.	
Includes:	Extend from Use case 8.	

Software Requirements Specifications

Use Case ID:	11	
Use Case Name:	Remove a game	
Actors:	Teacher	
Pre-conditions:	The game is already exist.	
Flow of events:	**User Action**	**System Action**
	1- Teacher should sign in and choose his type and category.	
		2- System displays all the games in the chosen category.
	3- Teacher choose to delete a game.	
		4- System delete this game.
Exceptions:	**User Action**	**System Action**
	1- Teacher choose to delete the last game of its category.	
		2- System displays error message that the category will be also deleted.

Software Requirements Specifications

Use Case ID:	12	
Use Case Name:	Show top 10 players of game	
Actors:	System	
Pre-conditions:	Students should play games.	
Flow of events:	**User Action**	**System Action**
	1- Student play games and choose to show top 10 in a game.	
		2- System displays top 10 players of chosen game.

Use Case ID:	13	
Use Case Name:	Provides Quality of a game	
Actors:	System	
Flow of events:	**User Action**	**System Action**
		1-Before the user play the game, System provides the user to choose the quality of the game.
	2- User choose the quality that he want.	

Software Requirements Specifications

Ownership Report

Item	Owners
Document Purpose and Audience, Software purpose, Software scope, Definitions, acronyms, and abbreviations.	Amira Sherif Alsayed
Functional and Non-Functional requirements.	Dina Essam El-Din
Use Case Model.	Youmna Essam Fawzy
Use Case Tables.	Nouran Mohssen El-Halawanny