

Group 5:

Yidi Wu

Iulian Toma

Ji Nan

Want to Be a Millionaire?

* URS

Table of Contents

[1. Introduction 2](#_Toc413111904)

[2. Functional Requirements 3](#_Toc413111905)

[3. Non-Functional Requirements 3](#_Toc413111933)

[4. Use case diagram 3](#_Toc413111933)

5. [Use-Cases: 4](#_Toc413111906)

[1. Register. 4](#_Toc413111907)

[2. Login 4](#_Toc413111908)

[3. Close Application 4](#_Toc413111909)

[4. Ready to Play 5](#_Toc413111910)

[5. Answer a Question. 5](#_Toc413111910)

[6. Get a ramdom question list. 5](#_Toc413111910)

[6. User Interface 6](#_Toc413111932)

Introduction:

In this project, we have to build small game program and the first step is to write the User Requirements Specifications. For this step we will provide the use-cases of all functionality that we can offer, a specification of user interface and also non-functional requirements. The objective of our application is to let two players against game.

In our game, two players can answer ten questions. If one of them answer the questions correctly more than the other, he will be the winner. And there is a textbox beside the question, it will update the result automatically after every question. Besides, after game is over, the players can choose to play again.

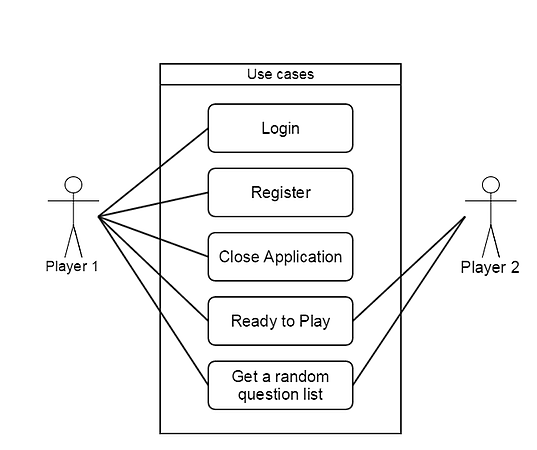
Functional Requirement:

|  |  |  |
| --- | --- | --- |
| Nr. | Description | MOSCOW |
| 1 | Playable by only 2 players | M |
| 2 | Internet connection required | M |
| 3 | Score displayed automatically after each correct answer | C |
| 4 | The game can be stopped anytime by one of the players | M |
| 5 | Answer time is maximum 1 minute otherwise system skips to the next question | C |
| 6 | When both players finish their game, systems shows the winner | M |

Non-Functional Requirement:

1. Only support Window 7, 8.
2. Not possible to play against computer.
3. No any update.
4. No any maintenance.
5. Game cannot be paused.
6. Using WCF.

Use case diagram:



Use Case:

1. **Register**

**Goal:** To register an ID for playing game.

**Pre-Condition:** Login form is shown

**Actor:** Player

**Trigger:** User left clicks register button in the login form

**MSS:**

1. The system shows the register form
2. User enters Player ID, Player name and password
3. User clicks register button

**Post-Condition:**  The system will turn to login form.

**Exception:**

**2a. User does not enter any one of information**

1. **Login**

**Goal:** To login in the game

**Pre-Condition:** Login form is shown

**Actor:** Player

**MSS:**

1. User enters Player Id and password
2. User left clicks Login button.

**Post-Condition:** The system goes to game form.

**Exception:**

**1a. User does not enter any one of information**

1. **Close application**

**Goal:** Application closed.

**Pre-Condition:** Application must be started and the current game is finished or in the process

**Actor:** Player

**Trigger:** User either presses “Game” “Exit” from the top menu or uses the “X” button from the right up side

**MSS:**

1. System shows the game result on the message box.
2. System closes the running application

**Post-Condition:** Program closed.

1. **Ready to Play**

**Goal:** Ready to play a game.

**Pre-Condition:** Players opened the game program or finished their game.

**Actor:** Player

**MSS:**

1. Player clicks on Ready button
2. Button will be disabled
3. System shows game starts.

**Post-Condition:** Game starts.

1. **Answer a question**

**Goal:** To answer and submit the question to the server.

**Pre-Condition:** Players finished their game an

**Actor:** Player

**MSS:**

1. Player chooses one of options
2. Player clicks answer button
3. System will submit the answer to the server

**Exception (Extension, Alternatives)**:

2a: Player clicks the answer button without choosing one of options

2b: Player cannot answer the question without clicking answer button

**Post-condition**: System will turn to the next question

1. **Get a random Question list**

**Goal:** Each player get different questions

**Pre-Condition:** Player Logged in

**Actor:** Player

**MSS:**

1. Player clicks on ready
2. System shows random questions on the screen

User Interface:





