## Use-Case 2: Start New Game: End Goal

### 2.1 High Level Description

#### 2.1.1 User end goal story:

When the user

Wants to start a new game

They press the ‘start new game’ button in the Pregame Scene

So that the application will start a new game or join an existing game without an opponent

#### 2.1.2 Event-response story

When the user presses ‘start new game’

It causes the application to ask the server if there are any games without opponents, if there are the user will join that game as player2 if not the application will start a new game and the user will be player1

So that a new game is either started or an existing game’s round 1 is completed

### 2.2 Trigger

The user presses ‘start new game’ from the Pregame Scene

### 2.3 Actors

#### 2.3.1 User

The user presses the ‘start new game’ button.

#### 2.3.2 Let’s Quiz Server – Ongoing games table

The application will ask the server if there is an ongoing game that needs an opponent and if so add the user to the game. If there is no ongoing games then the user’s game will be added to the Ongoing Games table and listed as ‘needing an opponent’

#### 2.3.3 Let’s Quiz Server – Questions table

The application will ask the user questions and check their answer

### 2.4 Stakeholders

#### 2.4.1 User

The user requires the app to run the game as expected

#### 2.4.2 Let’s Quiz Server

The device needs to be able to access the online database to check user details and current game state.

#### 2.4.3 User’s phone

The user’s phone needs to allocate memory and give the application the correct permissions to operate as required.

#### 2.4.4 Game Opponent

The other player, whether they have joined the game yet or not, is a stake holder. The user’s score, questions the user has been asked and what answers they gave will all effect the opponent’s game.

Whether or not the game was already ongoing or not there will always be an opponent who needs to compete against the user

### 2.5 Related Use Cases

#### 2.5.1 Start application

In this use case the application is already started, the user has logged in not using a guest account. This use case is dependent on the Let’s Quiz Server being in a useable state.

### 2.6 Pre-conditions

2.6.1 The user has started the application

* + 1. The has selected Play Game from the Main Menu

2.6.3 There is an open internet connection

### 2.7 Post Conditions

#### 2.7.1 Minimal guarantee

Give an error message to the user so they can rectify the problem, or upon an unrecoverable error the application will safely terminate and the user can restart it.

#### 2.7.2 Success guarantee

The application creates a new game or joins an existing game without an opponent, the user plays the round and the use case ends in the Post Game Scene.

### 2.8 Normal Flow

The use case begins when the user presses ‘Start New Game” in the Pregame Scene

|  |  |
| --- | --- |
| Actor | System |
| 1. The user presses ‘Start New Game’ button | 1. The app will connect to the Let’s Quiz server and check if there are any ongoing games without two players 2. The server response FALSE 3. The application starts a new game setting the user to Player1 4. The application asks the user a random question from the unasked question pool and provides 4 possible answers |
| 1. The user chooses the answer they think is correct | 1. The selected answer is checked to see if it is correct (TRUE) 2. The selected answer is highlighted in green, 10 points are added to the users score 3. The round timer reaches zero 4. The application ends the round 5. The Ongoing games table is updated with the results of the game 6. The Post Game Scene is loaded |

The use case ends.

### 2.9 Alternate Flows

#### 2.9.1 No connectivity to the Let's Quiz Server

If at step 2 of the normal flow the app cannot connect to the Let’s Quiz Server

|  |  |
| --- | --- |
| **Actor** | **System** |
|  | 2.1 A popup message alerts the user to the error  2.2 The application gracefully exits to the main menu |
|  |  |
|  |  |

#### 2.9.2 The serve’s response TRUE to the question ‘is there ongoing games without two players

If at step 3 of the normal flow when asked if there are ongoing games that need an opponent the server returns TRUE

|  |  |
| --- | --- |
| **Actor** | **System** |
|  | 3.1 The app asks the server for the details of the game requiring an opponent  3.2 The application starts a new game setting the user to Player2  3.3 The user is asked questions in order from the asked questions pool and 4 possible answers are presented |
|  |  |
|  | Normal flow will continue from step 6 |

#### 2.9.3 The user selects an incorrect answer

If at step 7 the selected answer is incorrect

|  |  |
| --- | --- |
| **Actor** | **System** |
|  | 8.1 The selected answer is highlighted in red, the correct answer is highlighted in green, 5 points are subtracted from the users score |
|  |  |
|  | Normal flow will continue from step 9 |

#### 2.9.4 The user selects an incorrect answer

If at step 9 the round timer has not reached zero

|  |  |
| --- | --- |
| **Actor** | **System** |
|  |  |
|  |  |
|  | Normal flow will continue from step 5 or 3.3 depending if the user is Player1 or Player2 |

### 2.10 Exception Flows

None

### 2.11 Key Scenarios

#### 2.11.1 Start a new game

2.11.1.1 The user selects ‘Start New Game’

2.11.1.2 The application queries the server if there are any ongoing games that do not have two players

2.11.1.3 The server responds FALSE

2.11.1.4 The application starts a new game marking the user as Player1 and Player2 as null

2.11.1.5 The user is asked random questions from the remaining question pool

2.11.1.6 The user answers the questions and is given a score as they answer correctly or incorrectly

2.11.1.7 The questions are stored as asked questions, the questions are removed from the remaining question pool

2.11.1.8 The round timer runs out

2.11.1.9 The application ends the round

2.11.1.10 The Post Game Scene is loaded

2.11.1.11 The game data is uploaded to the server

#### 2.11.2 Join an existing game that needs an opponent

2.11.2.1 The user selects ‘Start New Game’

2.11.2.2 The application queries the server if there are any ongoing games that do not have two players

2.11.2.3 The server responds TRUE

2.11.2.4 The server sends the game data for the ongoing game needing an opponent

2.11.2.5 The application starts a new quiz

2.11.2.6 The user is asked the questions saved in the game data as asked questions in order until they run out at which time the user will be asked random questions from the remaining questions pool

2.11.2.7 The user answers the questions and is given a score as they answer correctly or incorrectly

2.11.2.8 The round timer runs out

2.11.2.9 The application ends the round

2.11.2.10 The Post Game Scene is loaded

* + - 1. The game data on the server is updated

### 2.12 Other Quality Requirements

#### 2.12.1 Internet connection

The application needs to be able to communicate with the Let’s Play Server