Initial Requirement Model

VERSION INFORMATION

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author(s) | Description |
| 0.1 | 04/04/2018 | Michelle Vinall | Initial draft |
| 0.2 | 30/04/2018 | Michelle Vinall | Final Draft |
|  |  |  |  |

Contents

[1. Introduction 4](#_Toc512871692)

[1.1. Purpose 4](#_Toc512871693)

[1.2. Scope 4](#_Toc512871694)

[2. System-Wide Functional Requirements 5](#_Toc512871695)

[3) System Qualities 5](#_Toc512871696)

[a) External interface requirements (Non-functional requirements) – 5](#_Toc512871697)

[i) Usability 5](#_Toc512871698)

[ii) Reliability 5](#_Toc512871699)

[iii) Performance 6](#_Toc512871700)

[iv) Supportability 6](#_Toc512871701)

[v) Security 6](#_Toc512871702)

[4. System Constraints 6](#_Toc512871703)

[5. Assumptions and dependencies 6](#_Toc512871704)

[6. Domain Diagram 6](#_Toc512871705)

[7. Functional requirements 7](#_Toc512871706)

[a) Use Case Descriptions 7](#_Toc512871707)

[i) Use Case: End Goal: Launching Game 7](#_Toc512871708)

[ii) Use Case: End Goal: Login 7](#_Toc512871709)

[iii) Use Case: Login Registered users-Sub-function End Goal: Login 7](#_Toc512871710)

[iv) Use Case: Login with Facebook-Sub-function End Goal: Login 7](#_Toc512871711)

[v) Use Case: Login with Google Play Services-Sub-function End Goal: Login 8](#_Toc512871712)

[vi) Use Case: Play as Guest-Sub-function End Goal: Login 8](#_Toc512871713)

[vii) Use Case: End Goal: User Submit Question 8](#_Toc512871714)

[viii) Use Case: End Goal: Start a Game 8](#_Toc512871715)

[ix) Use Case: Choose Game Mode-Sub-function End Goal: Start a Game 8](#_Toc512871716)

[x) Use Case: Continue an existing game-Sub-function End Goal: Start a Game 8](#_Toc512871717)

[xi) Use Case: End Goal: Answer question 8](#_Toc512871718)

[xii) Use Case: End Goal: Facebook share 9](#_Toc512871719)

[xiii) Use Case: End Goal: Facebook Challenge/Invite 9](#_Toc512871720)

[xiv) Use Case: End Goal: Check Leader board 9](#_Toc512871721)

[xv) Use Case: End Goal: Check Achievements 9](#_Toc512871722)

[xvi) Use Case: End Goal: Exit application 9](#_Toc512871723)

[Full use Case Description for CCRD-Answer Question 9](#_Toc512871724)

[High Level Description 9](#_Toc512871725)

[Event-response story 9](#_Toc512871726)

[Trigger 9](#_Toc512871727)

[Stakeholders 10](#_Toc512871728)

[Related Use Cases 10](#_Toc512871729)

[Pre-conditions 10](#_Toc512871730)

[Post-conditions 10](#_Toc512871731)

[Minimal guarantee 10](#_Toc512871732)

[Success guarantee 10](#_Toc512871733)

[Normal Flow 10](#_Toc512871734)

[3.9 Alternate Flows 10](#_Toc512871735)

[Exception Flows 11](#_Toc512871736)

[Key Scenarios 11](#_Toc512871737)

[b) Use case diagrams 11](#_Toc512871738)

[i) Use Case: End Goal: Login 13](#_Toc512871739)

[ii) Use Case: End Goal: User Submit Question 13](#_Toc512871740)

[iii) Use Case: End Goal: Start a Game 13](#_Toc512871741)

[iv) Use Case: End Goal: Answer question 14](#_Toc512871742)

[v) Use Case: End Goal: Facebook Share and Challenge/Invite 14](#_Toc512871743)

[vi) Use Case: End Goal: Check Leader board/Achievements 14](#_Toc512871744)

# Introduction

The aim of the Let’s Quiz project is to design an online, multi-player, trivia question game that allows players to register or login using social media accounts or as a guest. It will allow social interactions such as Invite friends and share. The game will allow highest score recording and meaningful, fun game play.

We are going to create the application with the game engine Unity 3D, using C# as the primary programing language. Unity has many advantages for game development included extended support for 27 platforms. Unity has a game engine, an IDE and a user interface application all geared towards game development.

## Purpose

This Initial Requirements Model document will describe the requirements and specifications of the Let’s Quiz online trivia game. We will use this document to set the expectations for the development of this project. A requirements document is needed to guide the developers through to completion and should assist, our developers to define the intended functionality and parameters needed to develop this project.

## Scope

1. Produce an on-line single/multiplayer trivia game
2. Incorporate Facebook and Google Play Services SDK’s into game
   1. Facebook login
3. Facebook sharing
   1. Facebook inviting
   2. Google Play Services Leader board
   3. Google Play Services Achievements
   4. Google Play Services Login
4. Database integration
   1. Global high score
   2. Player statistics
   3. Managing game states of open games between players
5. Server integration
6. Finalize all implementations of game to be playable
7. Complete all tests to remove a bugs

# System-Wide Functional Requirements

1. F-1-Security Services
   1. Security services will be needed as we will be needing to authenticate users logins and will assist in these processes
2. F-2-Persistence Services

i) At the end of each round the application will upload game data to an online SQL database

c) F-3-Language Services

i) Language services will be integrated as English to start off with a hope of extending in future updates to support other languages

ii) Services that allow players with a disability would be beneficial and will be implemented wherever possible

d) F-4 Networking Services

* 1. Networking services will be required for multi-player playing or to login with Facebook or Google Services, and for the majority of interactions available in the game.

# System Qualities

The system qualities are known as non-functional requirements. These are attributes of the system and describe how the software will do a required task, not what the task entails. The FURPS+ acronym sums up the requirements F-Functional, and the URPS+ are the non-functional requirements

## External interface requirements (Non-functional requirements) –

### Usability

* + - 1. The user interface should be easy to use
      2. Interface should be compatible with mobile device screens
      3. The game mechanisms should be easy to learn, and navigate around.
      4. The users should be able to compete tasks in a reasonable amount of time

### Reliability

#### Availability

* + - * 1. The game should be available to players on request at least 99% of the time
        2. The application should have no more than 1 hour down time in any 2 month period

#### Accessibility

* + - * 1. Once the game is installed the user should have 24/7 availability of use.

### Performance

#### Response Times

* + - * 1. The users should be able to see a response from their interactions instantly
        2. For game launch the game should be playable within 30 seconds of launching optimum 10 seconds

#### Capacity

* + - * 1. The system should be capable of handling 100 users at any one time

### Supportability

#### Compatibility

* + - * 1. The game will need to be compatible with both Android and IOS devices

#### Maintainability

* + - * 1. The Game may wish to be added to in future updates so it will be advisable to begin with refactorable, clean code and thorough documentation

#### Documentation Requirements

* + - * 1. All required documentation will follow version control and all reporting protocols. We will supply all documentation necessary for the project

### Security

* + - 1. Security services will be needed as we will be needing to authenticate users logins and will assist in these processes

# 4. System Constraints

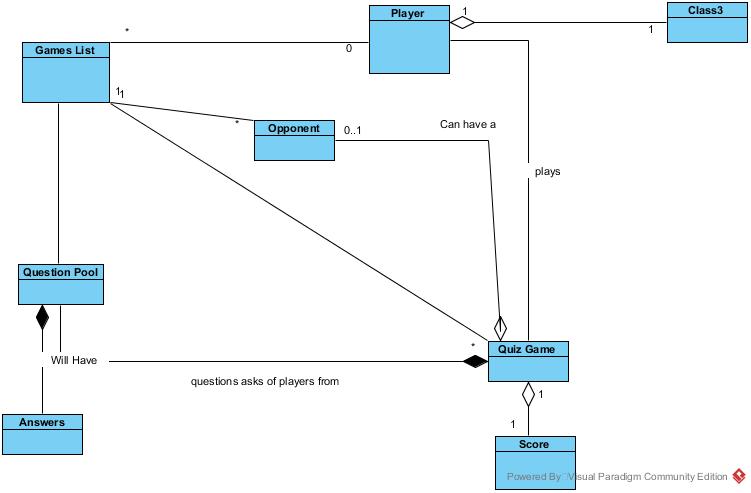
Constraints are the plus in FURPS+ and include

* 1. The game will require an internet connection
  2. Most of the coding will be done with C# and PHP
  3. The database will be in SQL.
  4. The user interface will have different levels of authentication handled by separate SDKs

# 5. Assumptions and dependencies

1. That the mobile device the application is installed on will meet the minimum system requirements.
2. It is assumed the user will have the technical ability to operate a touch screen
3. That all developers have knowledge of the required IDE and other necessary aspects including Unity 3d, Facebook SKD, and Google Play SKD.
4. It will be assumed that the Facebook and Google Play Services servers are available

# 6. Domain Diagram



# Functional requirements

## Use Case Descriptions

### Use Case: End Goal: Launching Game

When the user

Wants to start the application they click the application icon on their device

So that the application opens to show the login screen

### Use Case: End Goal: Login

When the user

Wants to login, they must choose login option

So that that the application allows login and displays pregame screen

### Use Case: Login Registered users-Sub-function End Goal: Login

When the user

Wants to play by logging in, they then press login

So that that the application opens to the login screen

### Use Case: Login with Facebook-Sub-function End Goal: Login

When the user

Wants to play by logging in with Facebook, they then press Facebook login

So that that the application connects to the Facebook Authentication server and allows login

### Use Case: Login with Google Play Services-Sub-function End Goal: Login

When the user

Wants to play by logging in with Google Play Services, they then press Google Play Services login

So that that the application connects to the Google Play Services Authentication server and allows login

### Use Case: Play as Guest-Sub-function End Goal: Login

When the user

Wants to play without logging in or first registering they press play as guest

So that the application opens to the pre-game screen

### vii) Use Case: End Goal: User Submit Question

When the user

Wants to submit a question they will press the submit question button

So that the application opens to the submit question scene

### viii) Use Case: End Goal: Start a Game

When the user

Wants to start a new game they will press the start new game button

So that the application will either start a new game or join an existing game

### ix) Use Case: Choose Game Mode-Sub-function End Goal: Start a Game

When the user

Wants to Choose game mode they close the game mode by pressing Corresponding mode

So that the application opens the correct game state

### x) Use Case: Continue an existing game-Sub-function End Goal: Start a Game

When the user

Wants to take their turn in a previously started game they press the games description

So that the application opens the correct game state

### xi) Use Case: End Goal: Answer question

When the user

Wants to answer a question they select the correct answer

So that the game can check the answer for correctness

### xii) Use Case: End Goal: Facebook share

When the user

Wants to share game they click share on face book button/link

So that the application connects to the Facebook server and allows sharing

### xiii) Use Case: End Goal: Facebook Challenge/Invite

When the user

Wants to Challenge/Invite they click the challenge button

So that so that the application connects to the Facebook server and sends invitation

### xiv) Use Case: End Goal: Check Leader board

When the user

Wants to check the leader board scores they press the leader board button

So that the application connects to the Google Play Services server to display the leader board

### xv) Use Case: End Goal: Check Achievements

When the user

Wants to check their achievements they press the achievements button

So that the application connects to the Google Play Services server to display their achievements

### xvi) Use Case: End Goal: Exit application

When the user

Wants to exit the application they press the back button on android and home button on IOS

So that the application closes down

# Full use Case Description for CCRD-Answer Question

### High Level Description

When the user

Wants to answer a question they select the correct answer

So that the game can check the answer for correctness

### Event-response story

When the user is ready to answer a question there must be a question displayed, either from a new or existing game and the round timer will start. They then select the answer they believe to be right which makes the program supply a new question to be answered. This repeats until timer ends, at which time correct answers are displayed in green and incorrect in red, as well as current score.

### Trigger

The user presses start game from the Pregame Scene

### Stakeholders

#### User

The user requires the application to supply questions promptly and correctly.

#### Let’s Quiz Server

The device needs to be able to access the online database to give new questions.

#### User’s phone

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

### Related Use Cases

None

## Pre-conditions

Application must be installed on mobile device

A new or pre-existing game must be opened

A question must be displayed.

## Post-conditions

### Minimal guarantee

Application will give an error to the user explaining existing problem or in the case of an unrecoverable error the application will safely terminate and the user can restart it.

### Success guarantee

The application continues to supply questions till round timer ends.

### Normal Flow

The use case begins when user gets asked a question

|  |  |
| --- | --- |
| **Actor** | **System** |
| 4. The user selects desired answer | 1. The application will display a question   1. Application starts timer   5. Score is added if correct   1. Application displays next question 2. Application ends timer 3. Application loads correct and incorrect answers 4. Application calculates and shows score |

The use case ends.

### 3.9 Alternate Flows

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

If at step 1 of the normal flow no question is shown

|  |  |
| --- | --- |
| **Actor** | **System** |
|  | 1.1 Question not displayed by application |
|  | 1.2 Application checks for server connection |
|  | 1.3 A popup message alerts the user to the server error |
|  | 1.4 The application exits to the main menu |
|  |  |

#### The user selects an incorrect answer

If at step 6 of the normal flow the user supplies incorrect answer

|  |  |
| --- | --- |
| **Actor** | **System** |
|  | 5.1 The incorrect 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 6 |

### Exception Flows

None

### Key Scenarios

User wants to answer question

Application retrieves an answer

User is given a score

Asked question is removed from applicable questions

Another question is randomly picked

Round timer ends

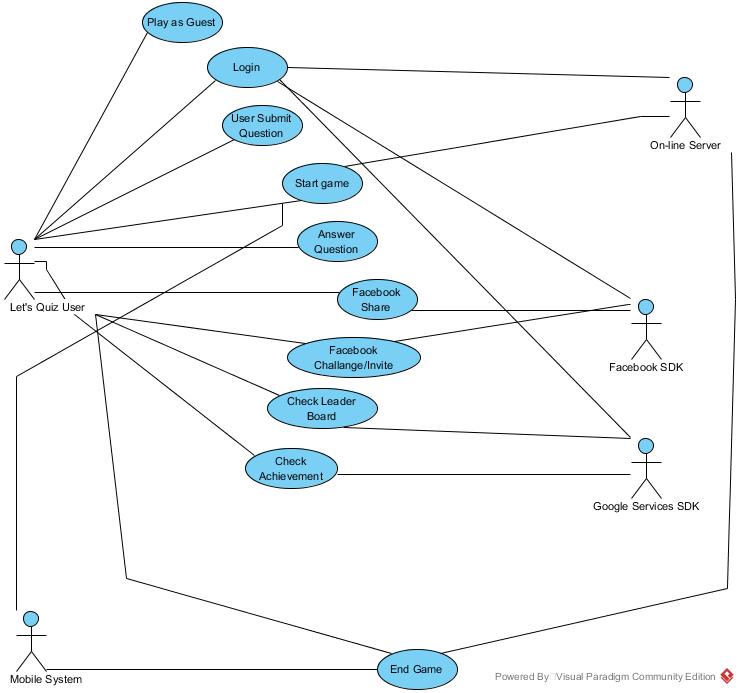
Scores are tallied and shown

Application ends round

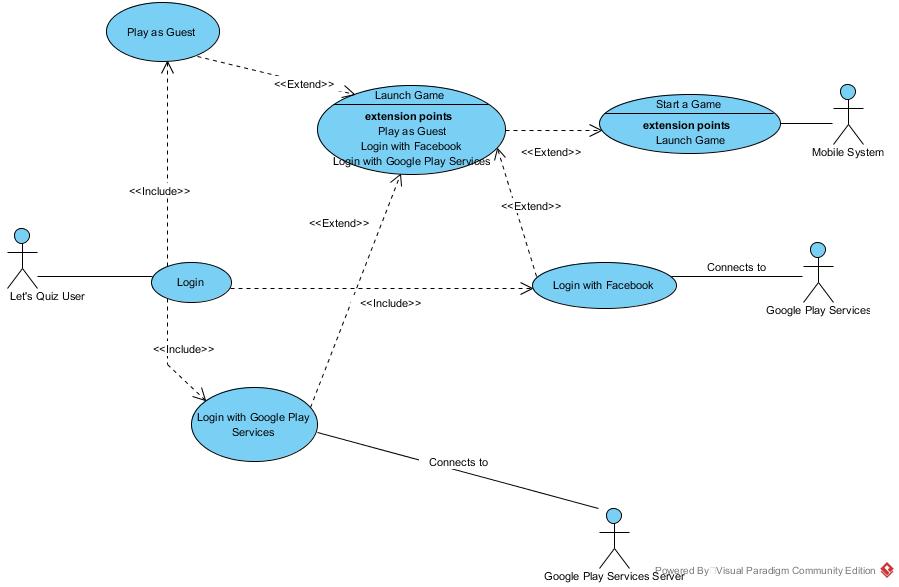
Game data is uploaded to server.

## Use case diagrams

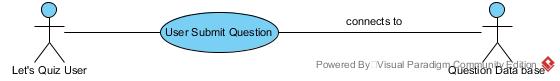
Basic outline use case application individual use cases underneath



### Use Case: End Goal: Login



### ii) Use Case: End Goal: User Submit Question



### iii) Use Case: End Goal: Start a Game



### iv) Use Case: End Goal: Answer question



### v) Use Case: End Goal: Facebook Share and Challenge/Invite



### vi) Use Case: End Goal: Check Leader board/Achievements

