|  |
| --- |
| Software Development Team |
| Let’s Quiz |
| Project Plan |

|  |
| --- |
|  |

Contents

[1 Introduction 1](#_Toc513728255)

[2 Project Organisation 1](#_Toc513728256)

[2.1 Project Introduction 1](#_Toc513728257)

[2.2 Project Team 1](#_Toc513728258)

[2.3 Project Team Rules 1](#_Toc513728259)

[3 Project Practises and Measurements 1](#_Toc513728260)

[3.1 Iterative Development 1](#_Toc513728261)

[3.2 Continuous Integration 1](#_Toc513728262)

[3.3 Independent Testing 2](#_Toc513728263)

[3.4 Project Configuration 2](#_Toc513728264)

[3.4.1 Unity3D 2](#_Toc513728265)

[3.4.2 Hosting Platform 2](#_Toc513728266)

[3.4.3 IDE 2](#_Toc513728267)

[3.4.4 Project File Structure 2](#_Toc513728268)

[4 Deployment 4](#_Toc513728269)

[5 Project Milestones and Objectives 4](#_Toc513728270)

# Introduction

The following document defines the implementation strategy of the development process for the quiz game Let’s Quiz. Here enclosed will outline the project organisation, practices and measurements, deployment strategy, and project milestones and objectives. Each section of the document will reference a specific topic as outlined above.

# Project Organisation

## Project Introduction

For a comprehensive introduction of the project, please see the project vision document.

## Project Team

The team is comprised of 4 members and 1 assessor;

* Aaron Peachy – Developer, Tester, Reviewer
* Collin Mckeahnie – Developer, Tester, Reviewer
* Charnes Nell – Developer, Tester, Reviewer
* Michelle Vinall – Developer, Tester, Reviewer
* Jim Tulip – Assessor

## Project Team Rules

For a comprehensive description of team rules, please see the team charter document.

# Project Practises and Measurements

## Iterative Development

The development of this project will primarily be carried out in iterations where specific project goals and critical software infrastructure will be implemented as outlined below. Each iteration will be a minimum of two-week blocks but can be longer depending on the tasks assigned. Each iteration will have its own unique plan, where the requirements are outlined, and tasks are assigned to team members. Each member is responsible for updating the iteration plan documenting their progress on their assigned task, however during each iteration meeting their progress will be updated if not done so already.

## Continuous Integration

As stated above, the development of this product will occur in iterative blocks where tasks are assigned to each member. The team will utilise version control through GitHub to share code and documentation. This will enable the team to have access to a product template through version control that was developed as a Technical Competency Demonstrator. Each team member will clone said template into their individual branch and work on their assigned tasks as outlined in the current iteration plan.

Once an assigned task is completed, the team member will create a review and pull request into the specified development branch and, the review will go through two stages, first through an assigned reviewer, then through the whole team, as stated in the iteration plan. Following the review, the pull will either be approved or rejected. If there are any merge errors detected by version control, they will be assessed by the team to determine the way in which to clear the merge error. Upon a successful merge, each team member must update their local development environment to assure they are working with the latest code.

## Independent Testing

For comprehensive coverage of how the product will be tested, please see the master test plan document.

## Project Configuration

### Unity3D

The frontend of the project will be developed using the Unity3D game engine, which is has a free edition that can be used by individuals and small-businesses. The software version that will be used by all team members is 2017.3.1. It is not the latest version; however, it is the version that the Technical Competency Demonstrator was built in. This game engine has the capability to generate a working game for the target platforms, as well as many others.

### Hosting Platform

The backend of the project will be hosted in Hostinger, where a dedicated MySQL server instance will be used to store game, and user information. For the frontend and backend to communicate with each other PHP will be used as a communication protocol.

### IDE

The project will be coded in MonoDevelop, which is the in-built IDE of Unity. The team has decided against using visual studio due to the majority using MonoDevelop.

### Project File Structure

The project structure that will be used within the Unity environment is as follows;

Assets/

|--Animation/

|-- Controller

|-- Clip

|-- Art/

|-- Sprite

|--Audio/

|-- Mixer

|-- Clip

|-- Editor

|-- Font/

|-- font folder/

|-- Contents

|-- Prefabs

|-- Scenes/

|-- Scripts/

|-- Controller/

|-- Data/

|-- Helper/

|-- Feedback/

|-- Integration/

|-- Facebook/

|-- Google/

|-- Object/

|-- Player/

However, if third-party assets will be used, the project structure will change to;

Assets/

|-- \_Game/

|--Animation/

|-- Controller

|-- Clip

|-- Art/

|-- Sprite

|--Audio/

|-- Mixer

|-- Clip

|-- Editor

|-- Font/

|-- font folder/

|-- Contents

|-- Prefabs

|-- Scenes/

|-- Scripts/

|-- Controller/

|-- Data/

|-- Helper/

|-- Feedback/

|-- Integration/

|-- Facebook/

|-- Google/

|-- Object/

|-- Player/

|-- third-party asset folder/

|-- Contents

|-- third-party asset folder/

|-- Contents

This will be used to make it easy to find files that are specific to the game and files that are specific to third-party assets.

# Deployment

The deployment strategy for the software is to publish the final product to the app store of both the iOS and Android platform. This allows for usage metrics, user feedback, as well as a method in which updates can be pushed and applied to all users of the product. This method is not guaranteed to work regarding applying updates, as downloading and installing the software will be at the discretion of the user.

# Project Milestones and Objectives

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Subject | Phase | Iteration | Dates | Objectives |
| ITC303 – Software Development Project 1 | Inception Phase | I-1 | 13/03 – 26/03 | Establish Vision  Establish Initial Use Case Model  Complete Preliminary Non-functional Requirement Analysis  Identify/Document Candidate Architectures  Establish Version Control |
| I-2 | 27/03 – 09/04 | Establish Risk List  Complete Full Description for Critical Core Risky Difficult (CCRD) Use Case  Implement Technical Competency Demonstrator  Create Test Plan  Establish Initial Project Plan  Complete Inception Phase Project Assessment  All relevant documentation be reviewed and pushed into version control, as well as updated in the wiki where relevant.  **Deliver Life Cycle Objectives Milestone (LCOM)** |
| Elaboration Phase | E-1 | 10/04 – 23/04  (Session Break) | **Database creation and UI draft**  Draft initial tables in database;   * Question * User * Game * Leaderboard   Draft initial UI for game;   * Splash screen * Registration / Login * Menu * Game * Result * Submit Question   **Question Pool Download**  Create UI to allow for registration and login  Refine user table in database where needed  Create PHP script for downloading questions from question table to establish communication between UI and database.  Create C# script to use PHP script and provide feedback to user on download progress.  Create tests for question download and user feedback;   * Is communication between UI and database established? * Are questions being downloaded? * Are the downloaded questions in the correct format? * Is the UI providing feedback on the download progress? |
| E-2 | 24/04 – 07/05 | **User registration and login**  Create UI to allow for registration and login  Refine user table in database where needed  Create PHP scripts to allow for user registration and logging in  Create C# scripts to allow for user registration and logging in using PHP script  Create tests for registration and logging in;   * Is user present in database after registration? * Can user log in with registered credentials that are stored in the database? * Is correct user information being retrieved after log in?   **Leaderboard**  Create UI to allow for user to view global leader board  Create PHP script to sort data in leaderboard table  Create C# script that displays data fetched from leaderboard table via PHP script  Create tests for leaderboard;   * Is data in the leadeboard table being sorted properly? * Is data being displayed properly in UI? |
| E-3 | 08/05 – 21/05 | **Single player playthrough**  Create UI for a game that display a question, i’s answers, and a time limit for each question.  Create UI for a game result that display the player score  Refine questions table in database where needed  Refine user table in database where needed  Create PHP script to submit player score to user table  Create C# scripts to display a question, its answers, time limit, and to use PHP script to submit score  Create tests for single player playthrough;   * Can a user start a new game? * Are questions being displayed properly? * Are the correct answers for the question being displayed? * Does the UI provide feedback on the correctness of the answer selected? * Does the UI update the score correctly according the correctness of the answer selected? * Does the UI update the timer correctly? * Does the result of the game get stored in the database? |
| E-4 | 22/05 – 02/06 | **Submit Question**  Create UI to allow for user question submission  Create PHP script to allow for user question submission  Create C# script to allow for user question submission using the PHP script  Create tests for user question submission;   * Is question present in the database? * Did the question upload correctly? * Is the question being downloaded correctly?   Complete any remaining tasks and resolve any issues  Complete Elaboration Phase Project Assessment  All relevant documentation and developed software be reviewed and pushed into version control, as well as updated in the wiki where relevant.  **Deliver Life Cycle Architecture Milestone (LCAM)** |
| Mid-year Semester Break | | | | |
| Subject | Phase | Iteration | Dates | Objectives |
| ITC309 – Software Development Project 2 | Construction Phase | C-1 | 10-07 – 23/07 | **Multiplayer Playthrough**  Create database structure to allow for multiplayer playthrough  Create PHP script to allow for communication between UI and database about multiplayer session  Create C# script to implement multiplayer playthrough and use PHP script to update database  Create tests for multiplayer playthrough |
| C-2 | 24/07 – 06/08 | **Social Media Integration**  Create database structure to allow for social media login  Integrate social media SDKS into project  Create UI to allow user to login via social media  Create PHP script to allow for social media login  Create C# script to allow for social media login  Create test for social media functionality |
| C-3 | 07/08 – 20/08 | **Push Notifications**  Integration Firebase into project  Create push notification layout  Create database structure to allow for push notifications  Create PHP script to allow for push notifications  Create C# script to allow for push notifications via PHP script  Create tests for push notifications functionality |
| C-4 | 21/08 – 03/09  (Session Break) | Contingency  Complete any outstanding tasks and resolve and issues  Determine test environment  Establish play test  Complete Construction Phase Project Assessment  All relevant documentation and developed software be reviewed and pushed into version control, as well as updated in the wiki where relevant.  **Deliver Initial Operation Capability Milestone (IOCM)** |
| Transition Phase | T-1 | 04/09 – 17/09 | Deploy application in trial environment  Complete first round play tests with participants  Resolve any identified issues  All relevant documentation and developed software be reviewed and pushed into version control, as well as updated in the wiki where relevant. |
| T-2 | 18/09 – 01/10 | Deploy Application in Trial Environment  Complete first round play test with participants  Resolve Any Identified Issues  All relevant documentation and developed software be reviewed and pushed into version control, as well as updated in the wiki where relevant. |
| T-3 | 02/10 – 13/10 | Contingency  Complete any outstanding tasks and resolve and issues  Complete Final Project Assessment  All relevant documentation and developed software be reviewed and pushed into version control, as well as updated in the wiki where relevant.  **Deliver Product Release Milestone (PRM)** |