|  |
| --- |
| Software Development Team |
| Let’s Quiz |
| Construction Phase Status Assessment |

|  |
| --- |
| Collin McKeahnie |

Contents

[1 Executive Summary iii](#_Toc524890527)

[2 Iteration Evaluation and Reporting iii](#_Toc524890528)

[2.1 Iteration 1 iii](#_Toc524890529)

[2.2 Iteration 2 iv](#_Toc524890530)

[2.3 Iteration 3 iv](#_Toc524890531)

[2.4 Iteration 4 v](#_Toc524890532)

[2.5 Iteration 5 v](#_Toc524890533)

[3 Discussion of Risks & Issues v](#_Toc524890534)

[3.1 Server Connection Unavailable v](#_Toc524890535)

[3.2 Project Scope v](#_Toc524890536)

[3.3 Game becoming boring v](#_Toc524890537)

[3.4 Cheating vi](#_Toc524890538)

[3.5 Interacting with foreign APIs vi](#_Toc524890539)

[3.6 infringing on existing IP and copyright vi](#_Toc524890540)

[3.7 iOS app submission vi](#_Toc524890541)

[3.8 Google Play Integration vi](#_Toc524890542)

[4 Progress Compared to Project Plan vi](#_Toc524890543)

# Executive Summary

The aim of the construction phase was to take Let’s Quiz from a single player game where users were asked a simple series of questions to a deeper and more complete experience.

Specifically, this would mean adding the following to the game:

* Multiplayer functionality
* Adding category specific rounds to the game
* Allowing for social media integration including Facebook and Google Play
* Adding push notifications so users were alerted when it was there turn.
* Dramatically increasing the scoring and leader boards and the global competitive nature of the game
* Adding offline redundancy and maintaining a single player mode of game play
* Publicly releasing the game on the Android Play Store and the Apple Apps Store

Overall, we were mostly successful in each of these undertaking with the two exceptions being Google Play social media integration and releasing the app to the Apple Apps Store. Detailed explanations of why these functions had to be removed from scope are outlined below but, in both cases, the simple reason is we could not meet the privacy requirements for these companies.

# Iteration Evaluation and Reporting

## Iteration 1

The primary goal for this iteration was to get the game to a state where users could play against each other. We knew this would be a large task and that it would primarily have to be completed by a single member of the design team.

The choice to elect a single team member to manage this entire task was essentially due to scaling as it was very difficult to break this task down into smaller pieces. The decision was made for Collin McKeahnie to carry out this task as he felt most comfortable with this part of the project.

By the end of this iteration multiplayer functionality had been added to the game and users could have multiple ongoing games happening at once. The only aspect that was not completed to a production level was in the UI where placeholder art work was used. It was expected that the U would not be at production level by the end of this iteration, the goal was only to have ‘multiplayer functionality’. There were two reason we decided not aim for production level completion of the UI in this iteration. Firstly, the time constraints of a single iteration would have made achieving any more very difficult and secondly at this stage it is expected the UI will go through several changes and to save double handling certain aspects of the design it was decided to leave the bulk of the design work until later in the phase.

There was a goal of having offline redundancy added to the game in this iteration. It was decided during the mid-iteration team meeting to drop this as a goal as it was dependent on aspects of multiplayer being functional, specifically a pregame check for opponents and the Game Lobby scene.

Overall this iteration was a success.

## Iteration 2

The goal of iteration 2 was to implement social media integration, push notifications to users’ phones when it was their turn and to extend the leader boards to have top questions and top question submitters.

It was a mistake to try to extend the leader boards to have a tally of votes cast before 'Voting on Questions' was implemented in the UI as there was no actual way to cast votes. According to the Project Plan adding voting to the UI was not scheduled until iteration 4.

By the end of iteration 2 the UI for the high score boards was implemented as best as possible but could only be tested by adding voting values to the database manually. Dynamic updating of the leader boards would not be possible until the UI was updated. Since we could not test the leader boards the priority for completion was dropped, so the ‘most correct’ leader board was not implemented until iteration 3.

Push notification research was completed, and a unity project was created to test the functionality and implementation of Firebase within a Unity project.

The other aim for this iteration social media integration, were also both not achieved, however this was less surprising as introducing foreign APIs can be very fiddly and time consuming. The plan to implement both Google Play and Facebook into the game in a single iteration proved too ambitious and the decision was made to push Google Play integration to iteration 3.

Overall, we did not achieve our goals this iteration. Social integration was held up this iteration due to a hospital stay (personal reasons) as well as an update of the Facebook policies.

## Iteration 3

This iteration was deliberately left light for many of the team members as it was foreseen last semester that integrating the different API’s of Facebook, Google Play and Firebase, our push notifications platform, into the game may prove challenging. Our aim for iteration 3 was to finish what was left from iteration 2 which was integrating Google and Facebook and to implement category specific rounds.

Offline redundancy which was put on hold from Iteration 1 was added to this iteration. Now that the multiplayer and game lobby had been fully integrated offline play could be implemented. It was not completed due to a build-up of other more important work items. Offline redundancy was started towards the end of the iteration but not completed.

Push notifications integration was completed by the end of this iteration however there was a spelling error in that resulted in notifications not being sent from devices.

We encountered significant problems integrating Google Play. At the end of this iteration it was decided to remove the option for users to sign in with their Google Play accounts. Because of the issues with Google Play the integration of Facebook had to be pushed another interaction.

Introducing categories proved to be quite tedious. Throughout this entire project we have had various issues moving JSON objects between the database and scenes within the game, however once these issues were resolved categories were added to the game.

## Iteration 4

The main goals of iteration 4 were completing documents including the beta tester’s survey, publishing the app, testing and bug fixes.

Publication to the Google Play Store and Apple Apps Store were both more complicated than expected, ultimately the app was published to Google’s Play Store but was not uploaded to the iOS store. The iOS store has stricter conditions for apps and does not accept apps the are still in beta stage. The app will be exported as a .IPA so users can install on their devices.

Facebook login and sharing were finally completed in this iteration. While it was expected that interacting with the different APIs would be difficult the complexity of integrating the application with Facebook and Google was underestimated.

## Iteration 5

The team did not initially plan to have an iteration 5 but it was recommended to us to take the extra iteration and it turned out to be needed. Two things took up the entire iteration, bug fixing and completion of outstanding items.

Due to the nature of building an application in a development environment and then having the execution environment being so different some bugs were expected, however the size and quantity of the bugs surprised us all. It is not an exaggeration to say that almost all our functionality broke when the application was released to the Play Store.

Following the testing of push notification functionality, it became clear that no one was sending or receiving any notifications during gameplay and that the integration of firebase was slowing down the app considerably. This was due to where the method calls to the cloud messaging were happening. As a result, the calls were removed from the app all together and were placed on the server when certain calls to the database was made.

The other issue encountered was certain tasks had been marked as completed in earlier iterations when they were not in fact 100% complete, as well as making allowances for tasks that had been completed but were not marked as complete.

# Discussion of Risks & Issues

## Server Connection Unavailable

Initially it was a problem that if the server was unavailable when it was required the game would crash. We have mitigated this issue by having the game fall back to single player mode where it only uses locally stored data.

## Project Scope

Technically this project was quite a challenge, combined with the time restraints of this phase we had to request an additional two-week iteration for the construction phase. While this did help us get Let’s Quiz ready for beta testing we now have to compress our timeline for the final phase of the project. To meet the new deadline for the final phase of the project we have shortened our iterations from 14 days to 9 days.

## Game becoming boring

Every addition we have added to the game, from multiplayer to question submission, to the various leader boards to changing theme to sci-fi has been aimed at making the game more engaging for users. If we have done our job right this will not be an issue, but should it be pointed out as an issue in beta testing we will react accordingly.

## Cheating

At this stage cheating has been deemed to be a non-issue. While technically on an Android all a user would have to do is move the game to the ‘waiting’ state, ie page flip to another app, and the game will pause giving them an endless amount of time to look up the answer to the question we don’t think it will be a problem due to the trivial nature of the game. Essentially if you’re the sort of person who is going to cheat a game like Let’s Quiz, you have bigger problems in life.

## Interacting with foreign APIs

There are three APIs that we will have to interact with; Facebook, Google Play and Firebase (for push notifications). Knowing the size of the task the only programming task given to Michelle was implementation of the social media Aps for Google Play and Facebook. We also have allowed 2 iterations for a single programmer to implement the Firebase API.

## infringing on existing IP and copyright

We do not believe this to be an issue. While we have not sort legal advice on the matter the basic idea of asking questions about a topic is not new and as such we do not see copyright infringement an issue for us.

## iOS app submission

As noted in the User Manual, the only way to download the game to an iOS device is through XCode. Since Let’s Quiz is still in beta we are unable to submit to the iOS app store since they only accept finished games. Once the game reaches a 2nd or 3rd stage of beta we can begin the process to submit to the app store.

## Google Play Integration

Google Play Services has been deemed out of scop, as it was found to be incompatible with the other plugins, SDKS and unity editions being used. The current build of Google Play services caused many resolve issues, when added and although was fully coded would simply not work. It caused many conflicts with the other packages and as it was more orientated for achievements and leader boards, we made a universal decision to deem it out of scope and abandon this avenue.

# Progress Compared to Project Plan

Completed Not completed Not started/ abandoned

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteration** | **Objective** | **Programmer** | **Notes** |
| C-1 | Multiplayer playthrough | Collin | Completed on time. |
| C-1 | Create tests for multiplayer playthrough | Michelle | All planned test scripts were written on time |
| C-1 | Implement offline redundancy | Aaron | This was deemed a low priority objective and was not started until later in the phase. |
| C-1 | Create the Game Lobby to show users ongoing games | Collin | A functional but ugly game lobby was created on time. The final product and art work were not completed until later as functionality was deemed higher priorities. |
| C-2 | Social Media Integration (Facebook) | Michelle | All scripts were written, and buttons were connected, but needed to test functionality which was not possible due to the existence of game play problems |
| C-2 | Social Media Integration (Google) | Michelle | Implementation of google play needed to be postponed till a working build could be uploaded to google console all scripts were written out but no functionality was implemented |
| C-2 | Create tests for social media functionality, push notifications and leader boards. | Michelle | All test scripts written |
| C-2 | Implement Push notifications | Charnes | Push notifications proved as difficult as feared and were not completed in this iteration. |
| C-2 | Extend Global Leader Board to Support Top Question and Top Question Submitter | Aaron | It was a mistake to put this as an objective before 'Voting on Questions' was implemented in the UI. The high score boards were implemented but could not be dynamically updated as the game did not allow for users to vote on questions |
| C-3 | Extend game to allow for rounds to be about specific categories | Collin | Completed on time |
| C-3 | Create tests for category specific rounds | Michelle | All tests were written on time |
| C-3 | Implement offline redundancy | Aaron | Work commenced on the offline redundancy. Not completed |
| C-3 | Social Media Integration (Google) | Michelle | Implementation of google play deemed to be out of scope due to incompatability. |
| C-3 | Implement Push notifications | Charnes | Push notifications were successfully integrated into the project, sending and receiving messages from the firebase console was not resulting in any errors. Nor was sending and receiving messages from the app or unity editor. However, the erroneous behaviour occurred when game play specific notifications were meant to be sent and received. This occurred due to the calls being placed in the app and as a result where never called. Instead they were moved to the server and calls were made during certain database updates based on game play status. |
| C-4 | Social Media Integration (Facebook) | Michelle | Completed |
| C-4 | Publish the game to the Google Play Store | Col | Completed |
| C-4 | Publish the game to the Apple Apps Store | Aaron | Compiled XCode integration. Unable to submit to app store as game is still in beta and iOS app store does not take beta games. |
| C-5 | Finish implementing offline redundancy | Aaron | Completed |
| C-5 | Finish leader boards | Col | Completed |
| C-5 | Fix push notification bugs | Charnes | Completed |
| C-5 | Refine UI | Charnes | Completed on time |
| C-5 | Execution of all test scripts | Michelle | Completed on time |