BSc in Software Development – Year 3

Mobile Applications Development 2 Project

Part 2 – Implementation

You may change and tweak your design as you progress through implementation. Very often, the original idea may not be achievable due to either time or skill constraints discovered as you learn. You should document the changes in an additional document to be part of the submission.

Asset Creation

This project is not about creating a collection of assets. The emphasis is on the technical work underneath. You can use art assets for the background screens. If you are using assets for the characters in your game, then you can only use a sprite sheet – premade functionality is not permitted. You need to provide links for your art assets with your submission.

You can used TextMesh Pro for you UI assets, but any fonts that you use need to be referenced. You need to create the UI elements from scratch, so completed assets downloaded from the asset store are not permitted.

Game Requirements

The game must have the following components present:

- Menu System – Splash Screen, Introduction, Preferences, Scores, Tutorials
- Scoring System – A score/rewards system with different score values for different enemies, items or tasks that the player must deal with
- Multiple Levels – Player competence and confidence in the game must be challenged with new levels of increasing difficulty. These should have been documented in the design phase.
- Multiple types of obstacle/enemy. There are different levels, so each must accommodate different threats.
- Sound System – Appropriate sounds and music must be incorporated. You do not need to compose the music. You can import a piece, but it should work with the game and needs to be referenced.

The User Experience is a critical aspect of any game. It is an important factor for the implementation of the control system, the menus and the other game components in this instance. This includes:

- the ease of learning how to play the game
- the placement of controls
- the rate of increasing difficulty
- the speed of gameplay
- win/loss conditions
- the appropriateness of the control mechanics
- the use of colours and text elements on the user interface

Submission

April 2nd – In-class demonstration of the game, other members of the class can play and vote for the "People's Choice Award". You will need to be able to answer questions about the development of the game during the demo sessions.

Your GitHub repository should be documented in a PDF file that is uploaded to Learn Online.

The file should contain a list of references for art assets and any other components to the game that you have incorporated but not written. Marks are going towards **your** work on this project. If you simply collect assets and get them working together, then the mark awarded will reflect this.

If you did not use GitHub, then you should export the assets from your game and then create a zip file with the assets and the PDF file. The zip file can then be uploaded to Learn Online. It is **your** responsibility to ensure that the assets are correctly exported and uploaded.

Marking Rubric for Implementation

0 - 35%	35 – 75%	75 – 100%
A selection of the basic game requirements have been implemented to a basic level Game Implementation will achieve minimum functionality Game Implementation may contain some syntax and/or run-time errors Game implementation code is poorly documented and/or formatted Game implementation will contain basic features; application will not be tested properly Game implementation code will not follow applicable coding conventions	Game requirements have been included to an acceptable level Game implementation will achieve expected functionality Game implementation will not contain syntax and/or run-time errors Game implementation code will be reasonably commented and/or formatted Game will be tested to a reasonable degree Game implementation code will follow appropriate coding conventions	Game requirements have been implemented to an advanced level Game implementation will not contain syntax and/or run-time errors Game implementation code will be well commented and/or formatted Game will be thoroughly tested Game implementation of code will follow coding conventions demonstrating use of appropriate patterns Game implementation adds significantly in a positive way to the design submitted

Breakdown of module marks for each component

Game Design Document: 20% of the module mark

Game Implementation: 60% of the module mark

In Class Assessment: 20% of the module mark