

Feasibility Study Report

Angad Kambli, 19114041
Ayush Gupta, 19114017
Chirag Wadhwa, 19114023
Devanshu Chaudhari, 19114027
Shubham Bansode, 19114019
Varun Gupta, 19114087

Abstract

Following is the feasibility report for the Course Project of CSN-254.
The project being a platformer game made in Godot.

Contents

| | |
|---|----------|
| 1 The Customer | 2 |
| 2 Visibility Plan | 2 |
| 3 The task to be undertaken | 2 |
| 4 Preliminary Requirement Analysis | 2 |
| 5 Deliverables | 3 |
| 6 Process to be followed | 3 |
| 7 Plan outline | 3 |
| 8 Risk analysis | 4 |
| 9 Technical Requirements | 4 |
| 10 Conclusion | 4 |

1 The Customer

In the last report, we had assumed the customer to be an end user who would be playing the game. However, it doesn't make sense to have a visibility plan with that kind of a customer. So, we are assuming a game publisher as a dummy customer. The game publisher can be assumed to have given a contract for a game which will then be published by them.

2 Visibility Plan

Since we have a small team, communication within the team is very simple and weekly meetings are enough to keep everyone apprised of the progress and to distribute the further tasks. We are using GitHub for Version Control so, other than the weekly meetings, intra-team communications regarding specific issues or bugs can happen on GitHub itself. Since it's on GitHub, the code will continually be visible to all team-members. Consequently, all the code will be visible to the dummy customer. Also, as the issues get created and resolved, the customer will be able to see all of that on GitHub thus making the whole development completely transparent.

3 The task to be undertaken

The customer has tasked us with the development of a sci-fi 2D-platformer game to be released for Windows 7, macOS 10.11, Ubuntu 16.04 and newer versions of these operating systems and possibly on HTML5, so that the game can be played on any PC that has web browser supporting web assembly. The necessary features required in the game have been specified in the following section of Preliminary Requirement Analysis(4).

4 Preliminary Requirement Analysis

The final product is required to have the following features:

1. A 2D platformer game with robust and interactive mechanics, and a well designed, immersive sci-fi theme.
2. A good setup to create any number of new levels.
3. A dialogue system for showing instructions and moving the story forward.
4. A few levels to showcase the mechanics of the game.
5. Presentable User Interface

5 Deliverables

The customer has tasked us with developing the game as specified earlier. Along with prototypes/iterations as per process, we are required to deliver the fully developed game, ready for deployment as per the features specified in 4. The end goal would be to create robust mechanics along with all assets and sprites so that any number of new levels and story-lines can easily be added afterwards as needed. To summarize, the deliverables would be a game having the features as specified in Preliminary Requirement Analysis(4).

6 Process to be followed

The project is relatively small in size, with fairly stable requirements. As such, a lot of change requests in the requirements are not expected. Due to the small size of the project, the overhead required to develop multiple iterations would not be paid off, and so it is better to deliver a finished product. The phases themselves are all concise, and so there will not be a lot of idle time for the developers. Due to these reasons, we have chosen to use the iterative waterfall model for our project. However, we will not stick perfectly (which is quite evident from the fact that coding has started way before the feasibility study) to this model, and allow some overlap between phases.

7 Plan outline

Following is the approximate plan outline to be followed for completing the project.

| | |
|----------|--|
| 21 March | Setup repository and start coding |
| 10 April | Finish first iteration |
| 12 April | Feasibility Report and start work on second iteration |
| 19 April | SRS document preparation |
| 26 April | Design Document and second iteration finishes. Testing and reporting issues |
| 3 May | Final Product along with all issue fixes |

8 Risk analysis

Godot being an Open-Source engine, has a chance of having a bug that might be discovered late in the development which ends up putting a wrench in the works for the whole development. However, we can work around this by changing the required mechanics or changing the implementation accordingly.

Another thing that can happen is the game might not be compiling properly for a particular platform. This could be solved by raising a ticket on the Godot's repo itself and getting it solved as soon as possible. The export templates that Godot uses are well-tested and robust, so any problem would most probably be a minor one.

Another risk would be that the game might lag on low-end devices

9 Technical Requirements

For this project we will be using Godot, an open source game engine. Following are the hardware and software requirements to run this engine efficiently.

Any remotely modern CPU will be able to handle Godot fine. Minimum 2 cores should be sufficient for running the IDE running smoothly. Godot is relatively light on RAM. As we are developing a 2-D platformer game 4GB will do the job. For a 2-D platformer game, GPU is not a necessity, integrated graphics will be more than sufficient to handle the graphics. In terms of storage, Godot editor is very efficient, the editor binary itself is around 35MB uncompressed, though this depends on optimizations. Export templates go at around 400MB totaling up to 500MB, for the entire Godot Engine.

In terms of software, engine works fine on Windows 7, macOS 10.11 or Ubuntu 16.04 or later. Github for version control.

10 Conclusion

The project is definitely feasible in the given time-frame and the identified risks also have their workarounds thus assuring of a working product by the end.