Deliverable E:

Project Plan and Cost Estimate

Team 1
Engineering Design - GNG 1103 - Section B03

Isaac Jeaurond - 300062033
Luke Alessio - 300190192
Seyed Ibrahim Hosseini - 300131150
Tesnime Zribi - 8402463
Mathuraa Balasundaram - 300114404

University of Ottawa - Faculty of Engineering
October 28th, 2020

Abstract

The purpose of this report is to develop a concise plan of action to ensure the completion of 3 prototypes before their respective due dates. We will identify tasks that must be completed to ensure functional prototypes. In addition, we will estimate the costs associated with the production of the three prototypes.

Table of Contents

List of Figures

List of Tables

1.0 Introduction

After completing the second meeting with our client, EllisDon, the team was able to obtain new information concerning budgeting and clarification regarding logistical questions in reference to the project. Further, the team worked collaboratively together to outline an effective design for the first prototype, which we developed from three core global concepts from the prior deliverable. In terms of feedback from the client, we concluded that our most recent design is an effective model and we should work to incorporate additional features to produce the ideal solution for EllisDon such as a cloud system component. Deliverable E works to outline a strict project timeline for the rest of the semester in order to monitor our progress to this day and as well as the aspects we must complete to attain a successful application for our client. At this stage, our conceptual design has been finalized and we have begun work for the prototyping portion of the project. Our first step is to construct a prototype that incorporates all the primary requirements and from that will additional features be accommodated in prototypes two and three. This project plan and cost estimate document will encompass the following components:

1. A list of all the tasks which need to be completed, an estimated duration for each task, as well as who is responsible for each task.
2. A Gantt diagram, which includes all significant project milestones and all dependencies. (made on Trello using BigPicture)
3. A list of the significant project risks and our associated contingency plans to mitigate the critical risks that are *reasonably* likely.
4. An estimate of the cost for all components and materials which we will need for the three different prototyping deliverables.

2.0 Project Plan

2.1 Prototype 1

Due November 5th

After completing the second client meeting, the team reached an agreement to construct our first prototype in a simplistic scheme. Further, this version of the application will incorporate the critical requirements that our client mentioned.

TASKS:

- Scripting in Unity - i.e. implementing code structures that allows the user to spawn a specific scene when clicking on a button in the menu
- Determine a tutorial format and establish 1-2 options for demo purposes
- Construct distinct scenes (not necessarily all of them, merely a couple to have the ability to efficiently build the first prototype)
- Generate a functional main menu and in-game pause menu
- Assemble the application for mobile devices and begin testing of the product. Ensure the testing is done a few days prior to the due date to give leeway for bug fixes and solutions for other problems that come up.
- *Milestones*:
    - Ability to spawn an object from the main menu on a mobile device - completed by November 3rd
    - Completion of the first model and script. Ensure the application has the ability to spawn a room from the main menu on a mobile device - completed by November 3rd
    - Submission of prototype 1 - November 5th

2.2 Prototype 2

Due November 12th

For the second prototype, the team aims to fix any issues that the application ran into during the building of prototype 1. Additionally, implementations regarding additional features will be completed here along with building on the fundamental functionalities.

TASKS:

- Reflect on feedback regarding the first prototype and determine methods the team can execute to ameliorate and fix issues within it.
- Fix scripts (if necessary)
- Fix tutorial format (if necessary). If feedback indicates that the format prior was good, work on building additional tutorials or build upon the one we created.
- Create more scenes based on prototype 1 feedback. (again, doesn't have to be all of them)
- Modify menus (if necessary)
- Similarly to prototype 1, build the application and test it a few days prior to the due date to ensure time to fix any issues.
- *Milestones*:
    - Have a fully functional menu with all tutorials available - should be completed by November 11th
    - Submit prototype 2 - should be completed by November 12th

2.3 Prototype 3

Due November 26th

From client feedback in our second meeting, EllisDon mentioned particular elements that would prove useful such as a cloud based network that users can upload and access files in. This aspect can upsurge the quality of user experience. Additionally, feedback from prototype 2 will be analyzed to implement plans of action for areas of improvement. At this point, the menu and settings should be complete as well as the scripting.

TASKS:

- Reflect on feedback regarding the second prototype and determine methods the team can execute to ameliorate and fix issues within it. If the team can fix the issues, fix them (whether it's the scripting or overall user interface).
- Complete ALL scenes that are required for the application. (use the previously developed scripts to enable the spawning of the different rooms)
- Ensure that all features of the application are functional when building an apk file for mobile devices.
- *Milestones*:
    - Have a fully functional mobile application will all the rooms available from the main menu - should be completed by November 25th
    - Submit prototype 3 - by November 26th

2.4 Breakdown of Tasks

Table 1. List of Tasks for Each Prototype

Prototype	Type of Task (Process or Milestone)	Tasks	Task Owner	End Date	Estimated Duration (days)
Prototype 1	process	Scripting in Unity	Isaac	November 2nd	6
	process	Determine a tutorial format	Tess	November 3rd	5
	process	Construct distinct scenes (a couple)	Mathuraa	November 2nd	5
	process	Generate a functional main menu and in-game pause menu	Seyed	November 3rd	6
	process	Testing	Luke	November 3rd	2
	milestone	Ability to spawn objects	All	November 3rd	None
	milestone	Completion of first model + script	All	November 3r	None
	milestone	Submit first prototype	All	November 5th	None
Prototype 2	process	Analyze feedback from prototype 1	Everyone	November 9th	2
	process	Fix scripts	Isaac	November 11th	3
	process	Fix tutorial format	Tess	November 11th	3

	process	Create more scenes (just a couple more)	Mathuraa	November 11th	5
	process	Modify menus	Seyed	November 11th	3
	process	Testing	Luke	November 11th	3
	milestone	Have fully functional menu + all tutorials	All	November 11th	None
	milestone	Submit prototype 2	All	November 12th	None
Prototype 3	process	Analyze feedback from prototype 2	Everyone	November 16th	2
	process	Fix issues from prototype 2	Isaac	November 23rd	5
	process	Complete ALL scenes	Tess	November 23rd	6
	process	Test features for apk file	Seyed	November 25th	2
	milestone	Have fully functional mobile app + all rooms	All	November 25th	None
	milestone	Submit prototype 3	All	November 26th	None

3.0 Gantt Chart

In order to make sure that our team is organized and making progress, a Gantt chart has been created using Trello - BigPicture. The following chart makes it easier to visualize our progress as a team and shows the owners of each task so that we may reduce miscommunication within the team. The Gantt chart shows the due dates and structure for the remainder of the project. Some of the due dates may be changed, but the overall project work sequence should resemble what is laid out on this chart.

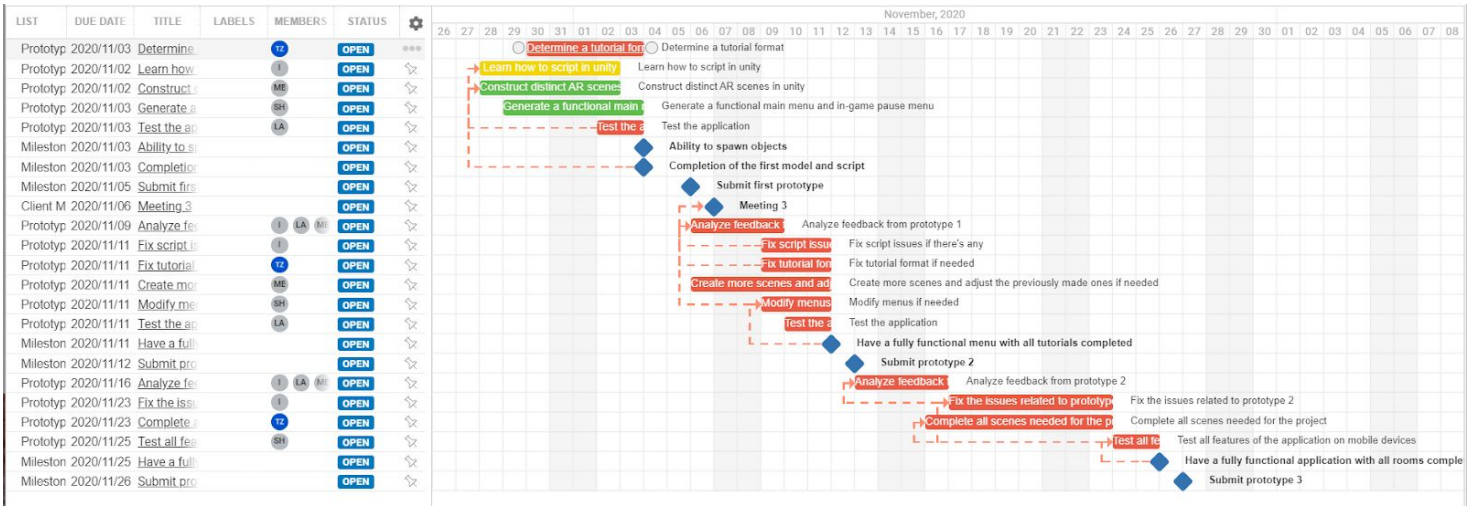

Figure 1. Gantt chart for the remainder of the project

The Gantt chart is used in conjunction with the Trello board, in which our team has a different, more detailed view of the project that allows for easier editing. Both the Gantt chart and Trello board ensure that all members of the team are kept up to date with important deadlines, and are able to visualize the project as a whole easily.

4.0 Risk Analysis and Contingency Management

This project is going to be created by Unity software, and all of the information we hold is saved within the server. A great risk that could impact our project is that if Unity crashes for any reason or if our files get corrupted and we can no longer access them. In this potential risk, unsaved work may be lost, which can negatively impact the strict timeline the team maintains to complete the project. Along with the loss of time, ideas the team has brainstormed may also be lost. In an event that this happens, our team has analyzed different methods that we can make sure our data is backed up and unwanted crashes will not create major delays in our timeline. Our main plan is to create a backup Google Drive folder which holds all of our project related documents, files, and information. We also planned to create local files in our personal devices to make sure that in an event that G-Drive is not available, we can access our project. In addition, members will save their work on their own computers periodically to ensure there is no loss of data and time in the project progress.

Another important part of our project is the funding we receive which our team has been focused on. Due to limited budgets, our team is working around ways to make sure that we have access to reliable resources while staying under budget. This means that we can risk the quality of our product, but fortunately we have created a table that takes into consideration all of our costs and where they are applied. We have already faced the issue of deciding which platform we will release our application onto since Apple Store will cost us all of our funding. However, the feature of uploading our application onto a store is solely optional and varies on the future endeavours the team would like to attain. Also in order to ensure that the team doesn't go over budget, the team will reserve a bit of money to prevent the risk.

A risk that our team could face is miscommunication, which we have managed to overcome by creating a server on Discord platform where we can communicate via text messages, audio calls, video calls, direct messages and voicemails. It became clear to us that miscommunication could cause significant delays in our work so we took the responsibility to make sure we are there for each other and we are actively updating the rest of the team on the progress that we have made. This platform has allowed our team to stay in touch with one another and help us connect and become a powerful team even though we've not yet met in person.

Another risk that will be taken into consideration is in the scenario of the application crashing with bugs or not operating efficiently in a smooth manner. In this case, the members that have the sole responsibility over that particular component will verify where the issue originates from and rebuild it accordingly, whereas, the member responsible for Unity scripts will test and fix the code with that section. Also, another team member will verify the operation of the entire system and feedback the issues they find with the team, then the team will work collectively together to solve the issues.

Finally, a risk that our team has considered is that the chance that one of our team members comes in contact with the *Sars-Covid-19* virus that caused the active global pandemic. In such an event, we know that a member falling ill will not only affect our progress on the project but also our motivation as a team and our mental wellness, which is why our team has agreed on active communication so that in such an event, we can continue our project in healthy and effective manners. Our plan is to make sure if a team member gets ill, we can divide the workload among the team in order to make sure our team member can have a stress free recovery!

5.0 Bill of Components & Materials

Table 2. Estimate of the Project Expenses

Expense Component	Expected Cost	Reasoning for Cost
Google Play registration	$25	In order to register an app for Google Play store, it would cost a one time registration fee. The fee will allow any user in Canada, USA, Australia and UK to download this application. If the application is needed in other countries than listed above, the team would have to pay more.
Apple Store registration	$99 per Year	In order for an application to be published in the Apple Store, the developer/developers will have to pay a fee of $99 USD per year. This cost will allow users from all around the world to download the application without other installation fees.
Labour costs *This expense does not apply in this project.*	$500 per developer	Since this is not a paid project, this cost will not apply, but this is a cost that we should be put into consideration. This cost was calculated to be at a rate of $14/hour for 36 hours, which is the estimated time of how much each member will spend on this project.
Unity AR Foundation, ARCore XR and ARKit XR plugins.	$0	Those 3 unity plugins are necessary for the making of an AR application. ARCore allows us to build applications for android devices while ARKit allows us to build applications for iOS devices.
Unity Personal Editor Software	$0	This is the program that we will be using in order to develop the application. The personal version is free and allows us to create as many scenes as we wish with all the assets available. This version is available to us as long as we don't make a revenue of more than $100,000 annually. If so, we must purchase the commercial version.
TOTAL	$0	Store feature is not a requirement but a future element we can incorporate.

6.0 Conclusion

All in all, the team is on a strict schedule and they maintain on track to attaining a fully functional application. The team aims to implement all required functions in addition to other features that will upsurge the user experience in order to provide the client with the ideal solution. This deliverable organized the remainder of tasks left to complete this application by Design Day, established a monetary breakdown, and a risk analysis and contingency plan. The summary of expenses, as of this moment, is approximated to be $0 due to the fact that every component required to build this application is offered through Unity, which is a free platform. In summary, the project will cost $0 depending on whether the team wishes to execute the plan of uploading the application on either Google Play, Apple Store, or both which will range from $25-$124. Three plug-ins have been determined for the application which equal zero, at the moment. Depending on the progress of the project in terms of prototype feedback, this may change.