Unity Project

Development Plan

Members: Anthony Jamora, Jonny Nabors, Thomas Burke

Draft 1

*CSC 4420*

*Unity Group*

*2015*

Table of Contents

1 Project Overview 3

2 Project, Purpose, Scope, Objectives 3

3 Team Organization 3

3.1 Team Leader 3

3.2 Specific Roles 3

4 Problem Resolution Policies 3

5 Project Plan 4

5.1 First Meeting 4

5.2 First Prototype 4

5.3 Phase 2 4

5.4 Phase 3 4

5.5 Project Completion 4

6 Configuration Management Plan 4

7 Technologies 5

8 References 5

9 Glossary 5

10 Document Revision History 5

11 Appendix 5

# Project Overview

The Unity Project is a collaboration between Wayne State University Computer Science and College for Creative Studies students to create a fully working, small form PC game known as a demo. By delegating tasks based off of the academic strengths of the two parties, both teams will be able to deliver a working product.

# Project Purpose, Scope, Objectives

The purpose of this project is to create a fully playable demonstration (demo) of the game for PC. This demo version will give players and potential investors hands-on experience with the product.

The demo will consist of one island in which the player will interact with a non-player controlled character who will spawn consistent waves of enemies for the player to face to demonstrate in-game combat. Additionally the player will be able to interact with elements in the environment such as doors and switches.

# Team Organization

## Team Leader

Jonny Nabors

To organize the project and follow up with other team members to ensure all goals are being met by the deadlines. The team lead will be the point of communication and the mouthpiece for the team to the client.

## Other Members

Anthony Jamora, Thomas Burke

Specific Roles: Though all team members will be spending time on across every area, each team member will focus most of their time working on their assigned specific areas.

# Problem Resolution Policies

Being a team of only three people, implementation of a strict problem resolution policy is key. In the event that a team member does not produce any work 12 hours prior to a due date, the team leader will reach out to the member in question to determine if any help can be spared. If there is no resolution 6 hours prior to the due date, the team lead will reach out to the instructor and TAs to make note of the incident. The other two members will then work to complete the assigned to the unresponsive team member to the best of their ability.

# Project Plan

Weekly meeting will be vital to capture the vision of the client. For now team members will meet on Wednesdays at the Detroit Public Library. Dates and times are subject to change.

## First Meeting, January 28

Discuss scope of project and overall goals including technologies, team dynamics, expectations, and schedule of deliverables.

## Phase One, January 28 - March 2

Create repository and project environment including assets, music, sound, and artwork. Begin implementing foundation of game mechanics such as player movement, environment, and events.

## First Prototype, March 3

Demonstrate functionally playable game with audio cues, lighting, enemy AI, and character movement.

## Phase Two, March 3 - March 31

Build on elements completed in Phase One including adding any missing character abilities as well as improved enemy AI and new environment assets.

## Second Prototype, March 31

Showcase stronger and more fluid gameplay mechanics along with fully explorable player world and.

## Project Completion

Full implementation of required game mechanics.

# Configuration Management Plan

The Unity project will utilize bitBucket, a free, online source control and repository management system. There will be a private repository available to group members, as well as to the client, their team, and the assigned TA.

# Technologies

Brief Overview of the technologies will be used for the development of this project.

## Unity 4.0

A video-game software suite offering robust game creation tools and debugging capabilities.

## MonoDevelop

Unity’s built-in text and scripting editor.

# References

|  |  |  |
| --- | --- | --- |
| Doc Number | Doc Version | Doc Name & Location |
|  |  |  |
|  |  |  |
|  |  |  |

# Glossary

**Example Term or Acronym** – Example Description

**Non-Player Character** - NPC

# Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Author | Changes |
|  |  |  |  |
|  |  |  |  |

# Appendix

Material including referenced documentation the web or elsewhere, as well as alternative designs or items/ideas for future improvements.