Jack Porter

[Email address]

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.   
When you’re ready to add your content, just click here and start typing.]

[Document title]

[Document subtitle]

Contents

# Analysis and Design

## Functionality and core requirements

## Flow/Class Diagrams

## Development Techniques

## Object Oriented Design

# Task Breakdown and Rational

## User Stories

We looked at the core requirements for this development product and at our basic game design that we had created and set about writing the user stories for this project. We thought about both the stories from the player’s perspective, as well as from the client’s perspective, trying to cover every one of the core requirements in our user stories and every aspect of the gameplay.

The user stories can be found below:

* As a Player I want to Move around so that I can navigate the level
* As a Player I want to Jump so that I can kill enemies by landing on them, help navigate the level

and activate buttons

* As a Player I want to Collect coins so that I can increase my score
* As a Player I want to Collect map pieces so that the chest will spawn
* As a Player I want to Avoid touching enemies so that they will not damage me
* As a Player I want to See enemies navigate the world so that the world feels more alive
* As a Player I want to Navigate around obstacles so that I can solve problems and have fun
* As a Player I want to Collide with the world and entities in the game so that the world is shown as a set of obstacles to navigate
* As a Player I want to Stay over 0 health so that my character doesn't die
* As a Client I want to Enemies to move between nodes so that it shows patrolling in order to make enemy movement more interesting to player
* As a Client I want to Have enemies make use of a finite state machine so that it breaks up behaviour into states making code cleaner and easier to debug
* As a Client I want to have the game be 3D and written with Object Oriented C++ using Visual Studio and DirectX so that the game fulfils the requirements
* As a Client I want to have objects in the environment have textures and lighting so that the game looks presentable while showing off technical features

## Work Breakdown Structure

Lewis designed the WBS for this project, using the user stories that we created as a group, he broke them down into tasks representing their smallest form that encompasses a module [APPENDIX]

## Critical Paths, Tasks, Timescale and Dependencies

## Grid Tasks and Times relating to WBS

## Testing Plans

I took on the lead role for designing and building the testing plan for this project, I have based the design around 2 types of black box testing, two types of white box testing and a section for logging ad-hoc testing

The black box testing that will be used is Systems testing, which makes sure that the program is meeting the functional requirements of the design and is the first level of testing of the product as a whole. We are also doing acceptance testing, which is used to test the product against the client’s requirements to make sure that they meet the design specified at the start of the project.

The white box testing that will be used is unit testing, which allows a developer to test specific functions or areas of code against a confirmed outcome to make sure that the code is performing as it should be. We are also going to include bottom up integration testing, which requires us to build and test each system from the lowest level upwards to make sure that each system will be working completely alone and with others before moving onto the next one.

The full test plan can be seen in [APPENDIX]

# Critical Reflection and Discussion of Group Work

## Evidence of Equal Distribution of Work

## Reflection of the Design Process

## Identification and Resolution of Problems

## Software Backup Methodology