Technical Design Document (TDD)

**Final Project – Non-Violent Rail Shooter**

Victor Haskins

Version: 1

Created: 10/26/2015

Last Updated: 10/26/2015

**Table of Contents**

Executive Summary 2

Project 2

Technical 3

Programs 3

Market Release 3

Time to Completion 3

Estimated Cost of Completion 3

Hardware and Software 4

2D Software 4

Sound Software 4

Programming Software 4

Development Plan 5

Milestones 5

Project Goals 5

Features 5

File Formats 6

2D 6

3D 6

Audio 6

Scripts 6

Scenes 6

Other 6

Asset List 7

Update Log 7

Executive Summary

**Project**

Utilizing work from a variety of earlier projects and then compounding upon that work, I am tasked with creating a rail shooter. Among the options available, I chose a non-violent rail game, where one might take pictures of and/or collect objects from the surrounding area as one moves through the game on a predefined path. The requirements say that one should have a mix of “free camera movement” and “fixed camera control.” If taking pictures, only a limited number of pictures should be available. If doing a collecting game, there needs to be a limited inventory space to store items. While more specifics will be detailed later in this document, the level must be set-dressed, with examples of saving/loading levels, using networking for multiple players, shader implementation, and threading.

**Technical**

Programs

The following programs will be required for the creation of Around the World:

|  |  |  |
| --- | --- | --- |
| *Program* | *Purpose* | *Cost* |
| Unity3d | Game Development Environment | $1500 |
| Audacity | Sound Creation | Free |
| Microsoft Visual Studio 2015 Community | Debugging Environment | Free |
| G.I.M.P | 2D Image Manipulation | Free |
| Blender | 3D Graphics and Animation | Free |

Market Release

This non-violent shooter is an in-house creation with no market release for demonstration purposes as a final project.

Time to Completion

The total estimated completion time for the non-violent shooter is five to six weeks. That includes time for coding and bug testing. It must be turned in by midnight December 11, 2015 for presenting to the class starting at 9:00am that day.

Estimated Cost of Completion

|  |  |
| --- | --- |
| Unity3d | $1,500 |
| 60 Hours Salary as this is the time I expect to use for one person. | $1,620.00 |
| **Total** | **$3,120.00** |

Hardware and Software

**2D Image Software**

|  |  |  |
| --- | --- | --- |
| Software Name | Description | Cost |
| GIMP | 2D image editing software if images or material is needed. | Free |
| Microsoft Paint | 2D image editing software if images or material is needed. | Free with Windows OS |

**Audio Software**

|  |  |  |
| --- | --- | --- |
| Software Name | Description | Cost |
| BFXR | Audio file creation and editing software if needed. | Free |
| Audacity | Audio file creation and editing software if needed. | Free with Windows OS |

**Programming Software**

|  |  |  |
| --- | --- | --- |
| Software Name | Description | Cost |
| Microsoft Visual Studio | Used to debug and create code | Free |
| MonoDevelop | Used to create prototype code | Free with Unity3D |
| Unity3D | Development Environ-ment. Used to release to multiple platforms. | $1500 |

Development Plan

**Milestones**

|  |  |
| --- | --- |
| Date | Milestone |
| 10/26/2015 | TDD Mockup Completed and turned in |
| 11/03/2015 | Level Design Constructed, Started with movement layout. |
| 11/10/2015 | Shaders built. Player interaction control started. Opposing AI’s started. Tools Started. |
| 11/17/2015 | Networking started. AI & Player controls mostly working. |
| 11/24/2015 | Threading Implementation Started. Basic Gameplay working. |
| 12/01/2015 | Demo Working, Tools @ 75%, and Debugging. |
| 12/11/2015 | Demo Complete and Submitted to Professor Fisher. |

**Project Goals**

Features

Checklist provided

* **Camera or Collection Device**
  + Delay (how long before allowed to repeat taking picture/object
  + Amount of Film/Storage Space
  + Zoom Level
  + Multiplier/Stack Storage
  + Ability to choose Camera/Backpack after choosing to start game, but before actually starting.
* **Creatures and Objects**
  + Each needs AI
    - Ex. “Wander around here” or “Move left and right”
    - Ex. Face Player/Run Away when pointed at.
    - Ex. Obstruct view…
  + Each should “activate” only with a certain distance from the player
    - Otherwise, do nothing
    - Also “deactivate” once moving outside the distance
* ***Tool/UGC*** – 4 tools: 1 external, 3 internal
  + Internal tools do not need to support exporting/importing
  + External tools need to support export/import of files to various locations
    - Need to follow User-Centered Design Concepts!
* **Engine** (*Internal Tool*)
  + Movements Tool from Projects 2 & 3
* **Level Creator** (*External – Map Creation only*)
* **Creature/Object Editor** (*Internal Tool*)
  + Tool to Create and Edit creatures or objects. Various options such as…
    - Type of AI creature or object should use.
* **Camera/Backpack Editor** (*Internal & External*)
  + Internal tool to add new cameras/backpacks to the game. These should be embedded.
    - Also allow player to change color of camera or backpack.
  + External tool to allow modder to create their own cameras or backpacks
    - Either c# form, direct text file editing, method similar to level creator
    - Modder will not have access to engine!
      * Only goal is to create new cameras/backpacks
      * Tool to export a file
* ***Specific Gameplay Elements***
  + **Save/Load** –
    - Include a scoring system. Ability to see these from menu
      * Scores saved w/ player name accociated with each score.
      * Store only 10 scores highest to lowest.
  + **Levels** – Each level set-dressed. i.e. needs to have creatures(or objects) static objects for player to see, and anything else that adds to the theming of the level. Programmer art acceptable as long as representations are announced.
    - One embedded level
    - One external level
  + **Scenes** – 4 scenes beyond level selection
    - Menu
    - Level
    - High Score
    - Camera/Backpack Selection
  + **Networking** –
    - At least two people need to connect to the game
      * Delay upon entering the game
      * Pick player option
      * Indicate ready
      * Begin
    - A 10 second minimum period where two players can see each other.
      * Note when this period starts in TDD
        + (ex. 20 seconds into start of game)
    - Extra Credit: chatroom before game start.
  + **Shaders** – two simple, basic shaders from scratch to use in this project
  + **Threading** – Your Creatures or objects should run on a separate thread from the rest of the game
  + **Analytics** – include solution to determine the following.
    - Percentage of each embedded cameras/backpacks use
    - Duration of each embedded cameras or backpacks use
    - Percentage of each embedded levels use
    - Completion rate of a single level (doesn’t matter which level)
    - Completion rate of each (all?) level(s?)
    - Average play time of the game.

File Formats

**Scripts**

|  |  |  |
| --- | --- | --- |
| Naming Convention | Description | Format |
| FilenameEditor | Editor script for an object script | .cs |
| FilenameDrawer | Editor script for an individual element script drawer | .cs |
| Filename | A script for an object | .cs |

**Scenes**

|  |  |  |
| --- | --- | --- |
| Naming Convention | Description | Format |
| SceneName | A scene in Unity3D | .scene |

**Other**

|  |  |  |
| --- | --- | --- |
| Naming Convention | Description | Format |
| TDD\_Project5\_Final | The final version of the Technical Design Document | .pdf |

Asset List

To Be Determined…

Update/work log