|  |
| --- |
| University Of The West Of Scotland |
| Design Document |
| Year 3 - Games Development Project |

|  |
| --- |
| Alasdair Hendry  9-26-2018 |

Contents

[1.0 – Proposed Plan 2](#_Toc526350666)

[1.1 – Development Team 2](#_Toc526350667)

[1.2 – Game Outline 3](#_Toc526350668)

[1.2.1 – Pitch 3](#_Toc526350669)

[1.2.2 – Summary 3](#_Toc526350670)

[1.3 – IDE 3](#_Toc526350671)

[1.4 – Project Plan 3](#_Toc526350672)

[1.5 – Risk Analysis 3](#_Toc526350673)

[2.0 – Game Design Document 4](#_Toc526350674)

[2.1 – Gameplay 4](#_Toc526350675)

[2.2 – Influences 4](#_Toc526350676)

[2.3 – Game Type & Research 4](#_Toc526350677)

[2.4 – Target Audience 4](#_Toc526350678)

[2.5 – Narrative 4](#_Toc526350679)

[2.6 – Graphics & Design 4](#_Toc526350680)

[2.7 – Level Design 4](#_Toc526350681)

[2.7.1 – Level Layout 4](#_Toc526350682)

[2.7.2 – Ambience 5](#_Toc526350683)

[2.8 – Audio Design 5](#_Toc526350684)

[2.9 – Moodboard 6](#_Toc526350685)

[3.0 – Technical Design Document 7](#_Toc526350686)

[3.1 – Development Approach 7](#_Toc526350687)

[3.2 – Target Platform 7](#_Toc526350688)

[3.3 – Hardware & Software 7](#_Toc526350689)

[3.4 – Mechanics 8](#_Toc526350690)

[3.5 – Development Environment 8](#_Toc526350691)

[3.6 – User Interface 8](#_Toc526350692)

# 1.0 – Proposed Plan

## 1.1 – Development Team

The development team for this project will only comprise of myself. Due to this, I will take on all responsibilities to ensure delivery of the finalised product, and will carry out the tasks for each role that a normal team would consist of.

In the interest of providing as much detail as possible, a table has been provided which outlines the roles & responsibilities that are included throughout the development cycle of a game.

|  |  |  |
| --- | --- | --- |
| **Roles & Responsibilities** | | |
| **Role** | **Responsibility** | **Team Member** |
| Project Manager | Takes full responsibility for the project and ensures the final product is delivered to the client, whilst meeting the initial brief. Oversees the day-to-day tasks of each team member and provides support where available. | Alasdair Hendry |
| Lead Programmer | Stitches together each code snippet provided by other team members, whilst working on the more advanced mechanics of gameplay. As this is a one-man team, there are no other programmers. | Alasdair Hendry |
| Lead Designer | Ensures the overall quality of each stage of design, including Art, Audio, Level and Story. Is tasked with making sure each designer is taking the game in the correct direction with their contributions. | Alasdair Hendry |
| Art Designer | Creates 3D Models, textures, UV maps, sprites, textures and animations. | Alasdair Hendry |
| Audio Designer | Scores background music, sound effects and ensures all audio is designed to suitable level whilst also following the theme of the game. | Alasdair Hendry |
| Level Design | Creates the environment in which the game will take place. Ensures the placement of props & buildings are suitable to the genre of the game, and scaling is correct throughout each level. | Alasdair Hendry |
| Quality Assurance | Plays the game, whilst specifically looking for bugs. Suggests possible fixes for each bug and a step-by-step guide on how to reproduce the bug. | Alasdair Hendry |

## 1.2 – Game Outline

### 1.2.1 – Pitch

|  |  |
| --- | --- |
| **Title** | <<GameName>> |
| **Platform** | Windows (PC) |
| **Genre** | City-Building / Economic Simulator / Real-Time Strategy |
| **Theme/Setting** | Set thousands of years ago, when settlers were laying the foundation for their new towns. |
| **Unique Selling Points** | Most games in this genre are restricted with either a grid or hex-like layout. This is not the case in <<GameName>>, as the player will be able to build wherever they wish. |

### 1.2.2 – Summary

<<GameName>> starts the player off with only a few peasants, some tools and a blank map which is their canvas. The player’s overall goal is simple; build and grow. To do this, they must satisfy the needs of their citizens by providing ample amenities and sufficient basic resources. On top of this, the player will be able to control the rate of growth by modifying laws & regulations and by creating trade routes through their towns.

## 1.3 – IDE

The games engine used throughout this project will be Unity Version 2018.1. This version is not the newest, but was recently released and offers a wealth of new resources compared to its predecessors. Notable features include;

* ShaderGraph – Offers a node-based environment in which the user can create PBR Shaders.
* Post-Processing Stack – Not a new feature, but noticeably updated to consider World Space Volumes.

Unity was selected as a development environment mainly because it is the engine which I have the most experience using. Also, the engine and all of its features are extremely well documented, allowing you to quickly check the Functions and Parameters that belong to a certain component. On top of this, Unity has tremendous community support, with a dedicated forum

## 1.4 – Project Plan

## 1.5 – Risk Analysis

# 2.0 – Game Design Document

## 2.1 – Gameplay

The player does not directly control any character within the game. Instead, they play as a governing body looking down on their citizens, and interact with them by selecting them individually and providing commands and jobs/roles.

The camera in the game will be slightly angled towards the terrain, and the player will be able to pan, zoom and rotate around the global Y axis using the mouse and keyboard.

Interacting with the environment is performed by simply using the mouse. The outcome will differ depending on what the player is specifically interacting with, however, most interactions will be gaining information on specific parts of lands or instructing citizens to perform work in a certain area. On top of this, the player will be able to interact with their town through a series of GUI panels which will contain town-related information.

## 2.2 – Influences

<<GameName>> takes influence from other successful games within the genre such as Anno 1404, Banished and Sid Meier’s: Civilisation series. These influences, however, only provide a basic foundation of features that will be built and expanded upon, and affect the gameplay style more than the art style. Other games, out with the target genre, such as Overwatch and Firewatch provide a source of artistic influence as the design leans more to the stylized end of the spectrum.

## 2.3 – Game Type & Research

## 2.4 – Target Audience

## 2.5 – Narrative

As <<GameName>> is not particularly story-driven, the player will most likely create their own narrative surrounding their town and its inhabitants as the progress through the game. As mentioned before, the gameplay follows the on-going development of the player’s civilisation as it grows from a petty settlement to a thriving metropolis, and the narrative is the actions that the player has taken to get it there.

## 2.6 – Graphics & Design

## 2.7 – Level Design

### 2.7.1 – Level Layout

Map design will remain consistent throughout <<GameName>>, using the same assets, theme and style regardless of the map location. However, the layout of each map will differ with each play-through. This is due to the fact that the map layout will depend on many variables, such as the difficulty that the player has selected, or if they opt to have a more mountainous terrain for that particular play-through. Allowing the player to have a slightly map dependant on their input will improve replayability and provides the player with more control over the experience they wish to have.

The player will be restricted to a play-zone that will be inset from the map border and they will not be able to move the camera outside of this zone. This allows for more terrain to be applied around the play-zone, which will look a lot more visually pleasing than simply having the terrain cut off at a certain point.

### 2.7.2 – Ambience

To create some ambience within the gameplay scenes, a day & night cycle will be added to the game to provide some extra diversity in the aesthetics department. Although this will not affect gameplay, the player will be able to toggle this mechanic off if they wish, as it may be distracting.

Ambient sound will also be used to add some extra life to the game. Sounds played will depend on the camera position relative to items such as fields, forests and marketplaces. On top of this, some sounds may change relative to the day & night cycle if the mechanic is enabled.

On top of this, in an attempt to add some extra detail to the game, there will be a focus on animating as many items in the world as possible. Something as simple as a waterwheel turning in the water can provide extra depth to a game. Lastly, a focus on small particle effects such as leaves falling from a tree will ensure that the player’s civilisation feels as alive as possible.

## 2.8 – Audio Design

## 2.9 – MoodboardC:\Users\b00330023\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Moodboard_Banished.jpgC:\Users\b00330023\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Moodboard_Firewatch_02.jpgC:\Users\b00330023\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Moodboard_Civilization.jpgC:\Users\b00330023\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Moodboard_Anno.jpg



# 3.0 – Technical Design Document

## 3.1 – Development Approach

## 3.2 – Target Platform

The final build of <<GameName>> will be for Windows (PC). The reason for this is directly correlated to the genre of the game as hardware limitations may hinder the experience on other lightweight devices due to the intense processing and graphical demand for a game of this scope.

## 3.3 – Hardware & Software

Creating a game from scratch requires many different pieces of hardware & software. This can be difficulty as it is important that each piece of software is on the appropriate version that is supported by other pieces. If version control is being used to develop between multiple devices, then even more difficulty is created as software versions must persist through each device. All the hardware & software used throughout this project is listed below.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Software** | | | | |
| **Name** | **Type** | | **Version** | **Use** |
| Unity | Games Engine | | 2018.2.2f1 | Provides useful libraries and API’s to increase production and efficiency during development. Also provides interactive interfaces for ease-of-use. |
| Visual Studio | I.D.E | | Comm. 2017 | Allows text editing for programming-related files such as .js or .cs, whilst providing text prediction and error logging based on the current language and engine. |
| Adobe Illustrator | Vector Images | | CS6 | Used to quickly create and modify vector graphics, allowing them to be scaled to any size and exported. |
| Adobe Photoshop | Pixel Images | | CS6 | Used to modify an image to suit a specific style or quickly create an alpha mask. |
| Audacity | Sound Editing | | 2.3.0 | Used to edit audio files, providing useful effects and modification tools. |
| Blender | Modelling | | 2.79b | Used to create 3D models and apply UV maps to existing objects. Also used to render promotional scenes. |
| Microsoft Office | Text Editor | | 365 | Used to edit text documents. |
| Github Desktop | Version Control | | 1.4.1 | Used to backup project versions and provides the ability to create branches of an existing project. |
| **Hardware** | | | | |
| **Name** | | **Use** | | |
| Keyboard | | Used to provide input to the engine and game. | | |
| Mouse | | Used to provide input to the engine and game. | | |
| Mid-Level Computer | | Runs the required software. | | |
| Monitor | | Displays the contents of the Computer. | | |

## 3.4 – Mechanics

## 3.5 – Development Environment

Unity is a development environment, available on Windows (PC), and MacOS. Used to develop many successful games, such as Hearthstone, Rust, Kerbal Space Program and Cuphead, it has become a must-have tool for any independent programmer (Dotan, 2015).

Built using a component-based foundation, Unity makes it easy for a programmer of any skill level to have a prototype up and running in a matter of days; possibly even hours. On top of this, the community for Unity is thriving with constant blog posts, documentation updates, YouTube tutorials and new versions being released every month.

Unity provide a wealth of API’s native to their engine, some of which include VR support, AR support, multiplayer, advert integration, team-based version control and cloud building. On top of this, Unity allows users to build their game to every major gaming platform, including Windows (PC), MacOS, Android, iOS, WebGL and Xbox One.

In contrast, Unity does have some major drawbacks. The most notable ones, when compared to engines such as Unreal Engine, are the lighting and shading systems. However, these only seem to be an issue out-of-the-box, and with a little knowledge a graphics programmer would be able to enhance the visuals of any Unity-made game.

Other flaws include the lack of stability in new versions, or versions in preview. The issue with this is that the most recent versions are so much better than previous releases that there is incredible incentive to update. Due to this, users may find themselves in difficult situations mid-development.

Comparing Unity to other engines such as Unreal Engine and GameMaker, Unity shines in most areas. Other engines don’t seem to offer the same features and support as Unity, and the documentation for them doesn’t seem to compare. Furthermore, Unity is very open-ended and is well-suited for many different project types, in contrast to other engines such as Unreal’s node-based FPS focus, or GameMaker’s strong focus on 2D.

Overall, Unity is the most easily-accessibly, lightweight and durable engine for any programmer. This combined with the incredible community support and previous experience are the main reasons that Unity will be used as a development environment for this project.

## 3.6 – User Interface

# 4.0 – References

Dotan, G. (2015). Top 10 Unity Games Ever Made. [Blog] SoomlaBlog. Available at: http://blog.soomla.com/2015/01/top-10-unity-games-ever-made.html [Accessed 3 Oct. 2018].