A video game screen with a robot and a planet

Description automatically generated

# **Technical Design Document for Populus Magos a god game by Mario Battiston**

*Game Title: Populus Magos*

Document version: v1.0

Written by (Mario Battiston)

Point of contact:

Mario Battiston

Contact Details:

Email- [st20234326@outlook.cardiffmet.ac.uk](mailto:st20234326@outlook.cardiffmet.ac.uk)

Date of publishing: 06/06/2024

Version number (v1.0)

Table of Contents

[**Technical Design Document for Populus Magos a god game by Mario Battiston** 2](#_Toc155178564)

[Development Requirements 4](#_Toc155178565)

[Development 4](#_Toc155178566)

[3d Modelling software: 4](#_Toc155178567)

[Photo editor (for textures): 4](#_Toc155178568)

[Pixel art editor used for the poster and the world map: 4](#_Toc155178569)

[IDE: 4](#_Toc155178570)

[Game Engine: 5](#_Toc155178571)

[2D/3D Software and API: 5](#_Toc155178572)

[Project Management 5](#_Toc155178573)

[Source Control 5](#_Toc155178574)

[Sound Software 5](#_Toc155178575)

[Project Structure 7](#_Toc155178576)

## Development Requirements

Development — IDE, test suites, editors etc.

3d Modelling software: *Blender 4.0*

* **Justification:** *Blender* was chosen as the 3d modelling software due to the Team having more experience with the software along with the support between Blender and Unity (The Game Engine) which makes workflow very effective. Well not as industry standard as 3dsMax, Blender is an amazing free 3d modelling software that proofed effective for this project.
* **Use:** 3d modelling, UV mapping, texturing, typology and generating textures for all Populus Magos’s 3d Model Assets

Photo editor (for textures):*GIMP 2.20.36*

* Justification: *GIMP* is a great little free photo editor that is 2nd only to photoshop. Due to the team’s familiarity GIMP was chosen to be used for any photo editing tasks.
* Use: Creating texture’s (Normal maps, specular highlights maps, Diffuse maps, roughness maps etc.) and editing UV maps.

Pixel art editor used for the poster and the world map: *Libresrpite*

* **Justifications:** *Libresprite* is a freeware pixel art editing software that can be used to make pixel art and can double up as a very basic photo editor. The Team has prior experience with this software and used it to help increase workflow when it came to art.
* **Use:** Populus Magos’s cover and the map texture that is used in game.

IDE:*Visual Studio 2022 & Visual Studio Code*

* **Justifications:** Visual Studio and Visual Studio Code are industry standard IDE’s that were used to make all the Games C# scripts that were then used in Unity (Game Engine). The familiarity as well as support for C# and Visual Studios was a boon to the project’s workflow. ­
* **Use:** Scripting in unity, debugging and C# Code.

Game Engine:*Unity 2022.3.9f1*

* **Justifications:** Unity is a titan of indie game development in the industry as well as game development in general. An easy-to-use game engine with tons of support both official and unofficial. Populus Magos (the game) is striving to be an Indie title and with Unity’s track record for successful games there is very few better places for Populus Magos to be made on.
* **Use:** Making the game, prototyping, game demoing, scripting, level design, importing assets, animation, and general game engine tasks

2D/3D Software and API: *DirectX 11.x*

Project Management — *GitHub and GitHub Desktop*

Source Control — *Unity Version Control*

Sound Software — Audacity and

## Asset Specifications

Details of supported asset formats and restrictions, for example model file formats, vertex winding, polygon counts or sprite file formats, width and height. Be sure to consider all required assets as appropriate to your game.

**Supported Asset Formats:**

* **Model file types**: *.BLEND* files were used when making the models in blender and *.fbx* is the file type each 3d model was exported as to be used in Unity. With *.fbx* armature, animation, mesh and texture data could be exported and uploaded onto Unity.

* **Texture File Types:** *.PNG and .JPEG* were the file formats used for the textures as well as any maps used for the textures.
* **Texture Width and Height:** Terrain Textures = 1024x1024. Character models and Building models all share the same texture the Imphenzia Pallet which was used to texture the models with colour and the pallet size is 512x512.

**Supported asset Restrictions:**

* **Model Polygon Count:** A strict poly count of 20 000. This is to keep the low polly feel of the game and impressive visuals will be handled by textures.

## Project Structure

**File Naming Convention** —Convention for naming code and asset files.

**Level / World Details** — Layout and asset list including appropriate size limitations.

**Development Plan** —Outline of the production plan from pre-production to delivery.

|  |  |  |  |
| --- | --- | --- | --- |
| Milestones | Date | Deliverable | Approval |
| Pre-Production End |  | TDD  Prototypes |  |
| Milestone 1 |  |  |  |
| Milestone n |  |  |  |
| Alpha |  |  |  |
| Beta |  |  |  |
| Final |  |  |  |
| Pitch and Play |  |  |  |

|  |
| --- |
| **Terminology:**  The terminology here is based on that used by Douglass in [Doing Hard Time](https://www.amazon.co.uk/Doing-Hard-Time-Developing-Addison-Wesley/dp/0321774930). Architectural design refers to the structural relationships between classes represented using Class and Package diagrams in UML. Mechanistic design refers to the interaction of runtime entities in performing particular behaviours represented using sequence diagrams. Finally detailed design is reserved for complex sequences of actions (Activity Diagrams) or state changes (State Chart Diagrams) performed by individual objects. |

* **Playtesting** — Overview of the kinds of tests, e.g. player control, combat, puzzle etc. and individuals responsible.