

Vision Document

“Fog of War”

Bachelor's Thesis 086
Sindre Haugland Paulshus

Date	Version	Description
21.01.2020	0.1	Initial setup.
24.01.2020	0.2	First draft.
31.01.2020	0.3	Edited product properties after a meeting with the customer. Added “options for additions” as 5.2.
03.02.2020	0.4	Added a descriptive line in 3.5 alternatives.
07.02.2020	1.0	Tidied up the references with Vancouver style. Added a description to the 4.1 sketch. Final draft.

Table of Contents

Table of Contents	2
Introduction	3
References	3
Summary of Problem and Product	3
Problem Summary	3
Product Summary	4
Description of Stakeholders and Users	4
Summary of Stakeholders	4
Summary of Users	5
User Environment	5
Summary of Users' Needs	5
Alternatives to Our Product	6
Product Overview	6
Role in the User Environment	6
Dependencies	7
Functional Properties of the Product	7
Crucial Properties	7
Options for Additions	8
Non-functional Properties and other Requirements	9
Documentation	9
Tools	9
Other	9

1. Introduction

This document describes the overarching vision for the project. This includes overviews of the problem, product, stakeholders, users and requirements. In short, the project consists of making a “Fog of War” (herby “FOW”) solution for the game Dwarfheim, which is being developed by Pineleaf Studio[1]. The FOW will be integrated into a larger project, the game itself. The project will be worked on by a single bachelor student for his bachelor's thesis during the first semester of 2020.

1.1. References

- [1] Pineleaf Studio. Dwarfheim [internet]. Pineleaf Studio; 07.02.2020 [updated 07.02.2020; 07.02.2020]. Available from: <https://dwarfheim.com>
- [2] Unity Technology. Unity [internet]. Unity Technology; 07.02.2020 [updated 07.02.2020; 07.02.2020]. Available from: <https://unity.com/>
- [3] Max Proude. Ultimate Fog of War [internet]. Max Proude; 02.12.2016 [updated 20.06.2018; 07.02.2020]. Available from: <https://assetstore.unity.com/packages/tools/utilities/ultimate-fog-of-war-76011>
- [4] Hukha. Lumbra [internet]. Hukha; 02.10.2018 [updated 07.01.2019; 07.02.2020]. Available from: <https://assetstore.unity.com/packages/tools/particles-effects/lumbra-2d-dynamic-lights-and-field-of-view-128759>
- [5] Brendan L.K. Fog Of War [internet]. Brendan L.K. 17.12.2015 [updated 09.12.2019; 07.02.2020]. Available from: <https://assetstore.unity.com/packages/tools/particles-effects/fog-of-war-51344>

2. Summary of Problem and Product

2.1. Problem Summary

Problem with	The FOW solution for the game Dwarfheim. It is a simple circle and does not feature line of sight.
Affects	The players who play the game, and game developers.
And as a result	Game immersion may be spoiled and developers have to design around the fact.
A successful solution will	Seem natural to the players and offer the developers new opportunities within the game's design.

2.2. Product Summary

For	Pineleaf Studio.
That	Wants a new FOW solution.
The new FOW	Is a better FOW solution.
That	Features line-of-sight, is modular and easily used by other developers.
In contrast to	the current FOW that is only a simple circle, and other solutions you can buy which are hard to integrate into the game.
Our product	Features line-of-sight and is created specifically for Dwarfheim, and as such integrated with the game from the start.

3. Description of Stakeholders and Users

3.1. Summary of Stakeholders

Name	Description	Role during development
Pineleaf Studio	The client. Represented by the Lead Programmer and the CTO. Their livelihood are dependent on the game's success.	Through representatives, Pineleaf Studio will guide the development and offer valuable feedback.
Developers	The other developers of the game. Their livelihood are dependent on the game's success.	May offer valuable feedback, knowledge and insight into various aspects of the game, design and programming.
Sindre H. Paulshus	The creator of the FOW solution. Dependent on the solution's success for his grade.	Will be developing the solution.

3.2. Summary of Users

Name	Description	Role in development	Represented by
Players	The ones that play the game when it is finished. Wants the game to be as good as possible.	May offer feedback on the product during development.	Themselves.
Developers	Developers of the game. Will be using, maintaining and possibly editing the source code after the product is finished.	Same as in 3.1.	Themselves.

3.3. User Environment

The solution must fit into their existing architecture for the game, using Unity [2] and C# to create it. It must fit into the game's aesthetic and work seamlessly. Source code will be reviewed and distributed using Git. The solution should be modular and not affect code of the larger project. It should also be well documented and tidy, so it can be used, modified and maintained by other developers after its completion.

3.4. Summary of Users' Needs

Need	Priority	Affects	Current Solution	Recommended Solution
Being able to see within a radius of friendly troops and buildings	High	Gameplay	Shader and a simple circle that unmask the map	Shader and dynamic area object that unmask the map
Line-of-sight: Terrain. Not being able to see through trees, walls etc.	High	Gameplay	None	Dynamic area object that stops if it meets terrain

Line-of-sight: Buildings. Not being able to see through enemy buildings	Medium	Gameplay	None	Dynamic area object that stops if it meets buildings
Visual representation of the Fog of War	Medium	Gameplay	A shown area and fadeout at borders	Same as current.
Grayscale of explored map	Low	Gameplay	Only of the static game pieces.	Grayscale map of the explored map with static and dynamic game pieces (enemies where last seen).

3.5. Alternatives to Our Product

A common thread between all alternatives is that it is a generic package solution, and would not be tailor-made for the game. As such, it could be hard to make it work as you want or expand or edit it if needed.

Alternatives	Description
Ultimate Fog of War [3]	A FOW solution sold privately on the Unity Asset Store.
Lumbra [4]	A FOW solution sold privately on the Unity Asset Store.
Fog Of War [5]	A FOW solution sold privately on the Unity Asset Store.

4. Product Overview

4.1. Role in the User Environment

The product will limit the visibility and information available to the players. It will play a role in how the players interact with, view and play the game. See sketch below for how it is imagined the product will work.



The sketch shows the player's view in white and the fog in dark gray. Note that the view is actually a circle with a given radius. The middle icon is a player's unit and the mountain icons are terrain the unit cannot see through. As such, the terrain casts a shadow which limits the player's view.

4.2. Dependencies

Dependency	Description
Dwarfheim	The solution will be created specifically for Dwarfheim and as such be dependent on it to function.
Unity	The solution will be created in Unity for a Unity game. It can only be used with Unity.

5. Functional Properties of the Product

5.1. Crucial Properties

These are the main and most crucial properties of the product. These are expected to be fulfilled for the product to be finished.

Property	Description
Lighting up area around friendly	As a part of the Fog of War, areas

units	around friendly units will have to light up to give the player sight.
Blocking view through terrain	The player's sight will be limited by terrain of higher height (walls, cliffs, mountains, trees, etc).
Blocking view through buildings	The player's sight will be limited by buildings, especially enemy buildings.
Only being able to see X tiles into the rocks in the mine	The game has a mine world with its own map and there is squared stone and ore as walls. The player should initially only be able to see the outer layer of the rock. This should be customizable, so for example after an upgrade they might be able to see 2 or 3 tiles.
Performance	The product should have little or no impact on the game's performance (frames per second).

5.2. Options for Additions

In the case of all the main properties being finished with ample time left, these are options for additions that would enhance the product further.

Property	Description
Grayscale of static objects map	Areas the player has no sight in will be grayscale, but still show static objects such as terrain, trees and rocks.
Grayscale of explored map	Areas the player has no sight in will be grayscale. If the player has been there at one point during the gameplay and there was enemy buildings there, those will remain in grayscale. Note that the grayscale image will not update even if the building changes. To update the player has to explore it again.
Height based Line of Sight	The player cannot see playable areas of higher height than their own. Contrary, the player can see

	playable areas of lower height than their own.
High ground sight	The player's sight is extended when viewing lower areas than their own.
Delayed vision loss	Upon losing vision (ie. a unit or building is destroyed), the vision will linger for a time before fading out.

6. Non-functional Properties and other Requirements

6.1. Documentation

- Vision document
- Requirements Document
- System Documentation
 - Wiki
 - Commented source code

6.2. Tools

- The development of the solution must happen in Unity 2019.3.0f3 with C#.
- Code and file distribution will be done with Git and Azure devops.
- Coding will be done with Visual Studio Code.
- Visual Studio Code will need the extensions "C# XML Documentation Comments" and "Azure Repos".

6.3. Other

- Aesthetically pleasing visual representation.
- Submission of Main Report and all linked documentation.