

AI Maze

20.06.2023

**─**

Gavin Lampe

# 

# Changelog

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Changes** |
| 1.0.0 | 20/06/2023 | Initial Setup |
|  |  |  |
|  |  |  |
|  |  |  |

# Contents

[Changelog 2](#_Toc1)

[Contents 3](#_Toc2)

[Introduction 4](#_Toc3)

[Rationale 4](#_Toc4)

[Background 4](#_Toc5)

[Terminology 4](#_Toc6)

[Proposed Design 4](#_Toc7)

[Non-Goals 4](#_Toc8)

[Software and Hardware Requirements 4](#_Toc9)

[System Architecture 5](#_Toc10)

[Data types 5](#_Toc11)

[Interface/API/Namespaces Definitions 5](#_Toc12)

[Which namespaces (Includes) did you include in your project? 5](#_Toc13)

[What functionality did each namespace provide to your code? 5](#_Toc14)

[Risks 5](#_Toc15)

[Alternatives 5](#_Toc16)

[Pseudocode 5](#_Toc17)

[System Pseudocode 5](#_Toc18)

[Evaluation 5](#_Toc19)

[Reflection 5](#_Toc20)

# 

# Introduction

In the TDD. Which namespaces (Includes) did you include in your project, What functionality did each namespace provide to your code.

## Rationale

/ What are you trying to accomplish? /

## Background

/ Describe any context that would be needed to understand this document, including any considerations. For example, what is expected? What is the purpose of what your making? /

## Terminology

/ If the document uses any special words or terms, list them here. For example, what does Agent mean? What does Area Modifier mean? This section is for terms you will use fill this in after you make your document. /

## Proposed Design

/ Start with a brief, high-level description of the project. The following sections will go into more detail. For example, summarize what it is you are needing to make. /

## Non-Goals

/ non-goals are stretch goals you personally have for the project; this includes anything that isn’t in the brief that you think you need to cover. /

## Software and Hardware Requirements

/ A list of all software being used, their versions and costs, as well as the targeted hardware constraints. Considerations should include what platform are you releasing to? /

# System Architecture

## Data types

• Integer – Used to store whole numbers such as 1, 256, -10

• Float – Used to store numbers with a decimal place such as 1.596874 and 0.2

• Char – Used to store single alphanumeric characters such as “A”, “b”, or “&”

• String – Used to store a group of alphanumeric characters such as “hello!”

• Vector2 – Used to store a set of co-ordinates for X and Y planes; co-ordinates stored as floats

• Vector3 – Same as Vector2 but also includes co-ordinates for the Z plane

• GameObject – A data type for storing Gameobjects which are the base entities of a Unity scene

• List – A list is an ordered collection of elements which can be any data type.

• Array – An array is a collection similar to a list except that it has a predefined size, making it less flexible but more efficient.

• Dictionary – A dictionary is also a collection but each element has a key that maps to it’s value, like giving a name to each item. That key can be used to retrieve and assign the element.

## Interface/API/Namespaces Definitions

/ Describe the various components and libraries you will be using that are inbuilt into unity. For example, GameObject, Image, SceneManagement, UnityEngine…etc. Link the Unity API Manual to show where to find information on those elements. /

### Which namespaces (Includes) did you include in your project?

### What functionality did each namespace provide to your code?

## Risks

/ If there are any risks or unknowns, list them here. Also, if there is additional research to be done, mention that as well. /

## Alternatives

/ If there are other potential solutions which were considered and rejected, list them here, as well as the reason why they were not chosen. /

# Pseudocode

## System Pseudocode

/ Written plan of the code and how you are going to write it in English. This is where you expand from what you were given in the brief. /

# Evaluation

## Reflection

/Provide a self-reflection on your performance. /