****

**BSc (Hons) Computing Systems**



**Dynamic cover-based system for Unity Game AI Players**

**Student Name: Leo Groza**

**Student Number: 888888**

**Supervisor: Tendai Mhlanga**

# Contents

[1. Introduction 2](#_Toc4022033)

[2. Present Values 2](#_Toc4022034)

[3. Obligations 7](#_Toc4022035)

[4. Risks 8](#_Toc4022036)

[5. Management 9](#_Toc4022037)

[6. Stakeholder 10](#_Toc4022038)

[7. Progress 10](#_Toc4022039)

[8. Accomplishments 10](#_Toc4022040)

[9. Issues 11](#_Toc4022041)

# Table of Figures

[Figure 1 – Project Gantt Chart 4](#_Toc4022152)

[Figure 2 – Project development environment. 5](#_Toc4022153)

[Figure 3- Nav-Mesh Pathfinder method 6](#_Toc4022154)

[Figure 4 - Edge-Detection via Field of View 7](#_Toc4022155)

[Figure 5 - Cover locations generated at implementation 7](#_Toc4022156)

[Figure 6 - Edge-Detection Code Snippet 8](#_Toc4022157)

[Figure 7 - Improved Edge-Detection method 8](#_Toc4022158)

# 

# 1. Introduction

The results achieved in the project will be outlined in the report by comparing the initial requirements with the results produced in the process to date. The report is beneficial to the project as it presents the accomplishments done so far and what processes need to be completed to satisfy the requirements of the stakeholders and to fully complete the initial requirements of the proposed project. The report provides insights to the development process of the project in different areas such as involvement of the stakeholders, requirements management, stakeholders involvement and development risks and risk to be encountered in the process.

# 2. Present Values

The progress of the project is going to be measured with the help of the Gantt Chart and the initial Project Plan submitted in the first semester. The report is to be constructed and contrasts are to be made in account with the Gantt Chart presented in the report.

Figure 1 – Project Gantt Chart

Minor changes have been added to the project scope in completion of the initial plan. The initial plan will not be affected as the changes added are set to be improvements or upgrades to the initial aims and objectives as a result of the continuous development of the game engine in use that is set to release new updates until June 2019, updates that will change minor views upon the current aims and objectives.

The submission date for the project is on the 10th of May 2019 and the amount of research and development in progress and to be carried out is vast, it is a huge process but not an impossible one. A number of requirements have already been completed since the beginning of the project. The stages in the development process have not been followed as presented in the Gantt Chart. Some of the stages have been skipped to be returned to in a later development stage.

Before beginning the actual research in *multi-threading* and *Unity Jobs System*, the first stages in the development, a turning point of the process was to set up a development environment. The environment was planned to be constructed in a later stage of the process, considering the implications that may arise while tempering with the game *Patient Zero* which is still in development.

At the end of the project, the result would have been integrated in the game, results that would have been developed in an isolated environment, apart from the game, containing assets and characters taken from *Patient Zero*. The conclusions following discussions with the stakeholder states that the best results would be obtained by working in a similar environment with the game or within the game itself. For this matter a game scene in construction from *Patient Zero* has been chosen as the development environment for the project. Every action, upgrade, modification and progress measurement during the process will be carried out in the designated scene.

The scene has been minimalized for development purposes to contain as few elements as possible. Every asset in the scene is implemented to fulfil a development purpose and will stand as aids to enhance the environmental abilities of the AI pictured in the lower left corner. During game play the AI will make use of all the modifications, upgrades implemented during the development process such as *Filed of View, Ray Cast* and *Dynamic Pathfinding* to construct and follow a safe path towards the target.



Figure 2 – Project development environment.

Following the set-up of the development environment a number of project objectives, present in the Gantt Chart have been completed to date. Basic C# and Unity Game Engine programming and development skills have been acquired for project development, until this point, the skills can only be called basic but must improve in future development.

A research has been conducted to feature the pros and cons of the *A\* Pathfinder* method and to gain insights in the functionality of the method, to upgrade or find the best implementation where the method can provide the best functionality. In total there are 64 *Pathfinder* methods that can be implemented in games software from which approximately 24, the most common ones, have been researched. Each of the methods has a purpose and it is specifically designed for s different type of game. The current method implemented in *Patient Zero*, *A\* Pathfinder,* is the most suitable one as it provided to date the best game paly rendering process.

The basic *pathfinder* method implemented in Unity is called a *Nav-Mesh. Nav-Mesh* is a data structure designed to find a path between some objects in a large area. By following the *Nav-Mesh* implementation and upgrade the scripting to benefit our project, the result is a more dynamic and less resource consuming pathfinding method compared to the one implemented previously in the game.



Figure 3- Nav-Mesh Pathfinder method

The implementation of the method has led to better rendering results according to *Unity Profiler* and a smooth gameplay. One of the minor changes in the project scope is to upgrade the implemented pathfinding method, *A\* Pathfinder,* with the less resource consuming method, *Unity Nav-Mesh.*

Progress has been made in upgrading the *Edge-Detection* method. The objective of the *Edge-Detection* method is to allow the AI to make use of the environment, to use locations in the game environment as *safe* or *cover locations*. The major issue to be first encountered was the need to prepare the game environment before the game start, which keeps the game play from being dynamic. Through scripting it was managed to keep the game dynamic and make use of the *Field of View* method to implement the *Edge-Detection* method. The result has been promising and the game was made dynamic.

A picture containing indoor, wall

Description automatically generated

Figure 4 - Edge-Detection via Field of View

Although the implementation of the method has been successful, the main drawback was the software would generate thousands of cover locations in the game environment and outside of it, for a yet unknown reason, and placing multiple cover location one on top of another in the same location.

A picture containing screenshot

Description automatically generated

Figure 5 - Cover locations generated at implementation

The large number of game objects generated by the software had an impact over the rendering process and follow up measurements were needed. A simple math logic, implemented in a code designed to eliminate as much doubles as possible from the cover locations, and implemented in the *Field of View* scrips led to developing a performant *Edge-Detection* method where improvements can actually be seen.

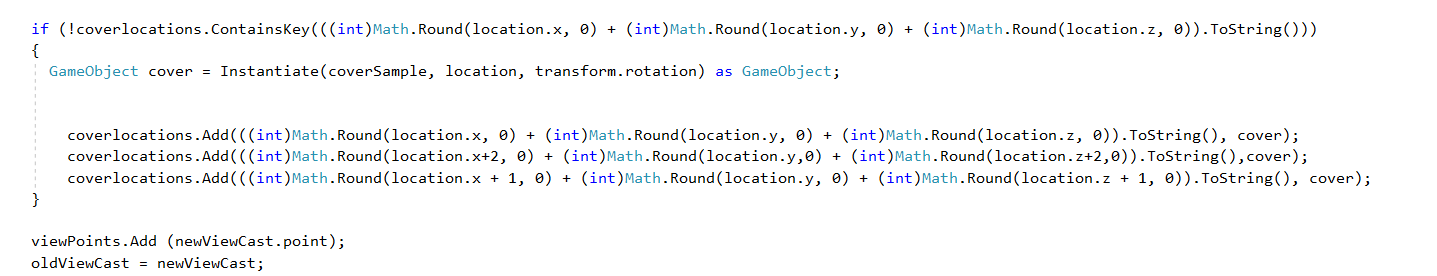


Figure 6 - Edge-Detection Code Snippet

A picture containing indoor

Description automatically generated

Figure 7 - Improved Edge-Detection method

Progress can be seen in different stages of the project but there is still need of work to be conducted in research and development until the findings can be recorded.

# 3. Obligations

The project development is running efficiently but behind the schedule. The late start of project development and complexity of the aims and objectives have an impact on the overhaul development. More time will be required to be allocated to the development of the project.

The main aims of the project are:

* Review Unity Jobs multi-threading system;
* Separate individual threads from the main thread;
* Allocate run time dynamic cover at for game objects and use Ray-cast edge-detection framework in the process;
* Develop AI game objects coordination framework.

Approximately 40% of the objectives have been reached but there is still a very large amount of research to be conducted and recorded. Extra time will be allocated to the project considering the fast approach of the submission date. Regular meetings are conducted between the stakeholder and me where we have established that not all the aims must be fully completed, as long as at least 50% of the aims are completed and progress can be observed it is seen as an achievement. The main focus remains the full completion of the project with significant progress to be seen. For this reason, more late hours will be added to the development process.

# 4. Risks

Risks represent a part of any development project. Failure in controlling, monitoring and removing the risks can lead to a standoff of the development process, or worst-case scenarios, to the sudden finish of the project. The risk assessment process is designed to identify the possible risk in project development and methods to combat those risks in the eventuality of their occurrence.

Similar to any software development process there are several common risks taken in consideration:

1. Lack of experience in project management which may result in failure of meeting the deadlines;
2. Data loss, system failure or work corruption;
3. Failing to acquire project resources;
4. Focusing on different priorities ahead of the project therefore spending valuable time necessary for implementation;
5. The final product is not satisfactory for the stakeholder.

Disregarding the Gantt Cart and the schedule is the first risk present in the development project. Quite a lot of time can be wasted on implementing insignificant features of the project without having enough time to meet this submission deadline. The project had a late start and this risk is present in the project at this moment, but extra development time will be allocated to combat this risk.

The second risk implement possibility of losing work or data due to hardware failure or corrupted data, but measurements have been put in place to combat this specific risk with regular data backup and hardware maintenance. There are several copies of the project on different storage devices and every 14 days another copy is made containing the progress made to date. 14 days is quite a long period of time to perform a backup but unfortunately the project is over 100 GB in size which takes up to 4 hours to copy.

The third risk refers to the failure of acquiring project resources. This risk refers to the game engine used in the development process. The development of the game on which the project is focused has begun three years ago. The game engine is in continuous development and every two months new features are released along with software updates. Software updates will deprecate old features and can temper with the existing development of the game, for a game begun three years ago, with software updates major modifications must be made to have the possibility of recompiling the game again. For this reason, the software update is blocked, the game continues development on an old version which denies access to the new feature releases. Some necessary, updated components have been added to the old version but the completion of one aim require an update which will be released before May 2019. Research is conducted to find a solution for implementation without affecting game functionality, if needed the game engine will be updated to the final version and modifications will be carried out to eliminate and update the deprecated features.

The fourth risk is concerned with priorities. It is quite difficult to balance family life, work and the development of the project. Having a full-time job and a one-year old son does not make things any easier in terms of project development. Sacrifices have to be made and after my son goes to bed, I spend most of my nights working on the project. Having an optimistic attitude and knowing that I will finish in two months’ time make things slightly easier so I work as much as I can thinking for the best.

The final risk of the project relates to the stakeholder and his satisfaction towards the final product. The risk cannot be known until the very end of the project, but regular meetings are carried out with the stakeholder and he has shown contempt regarding the development process in terms of programming and in terms of documentation his statement suggests that more progress must be made.

# 5. Management

The project is managed using the iterative methodology, this being the methodology chosen in the initial stage of the project. The iterative approach is suitable for early builds, progress measuring possibilities in early stages and testing in small iterations. The work on the project begun with modest implementations on the objectives defined in the requirement development stage of the project.

The iterative approach has been chosen for its advantages in the visualising early results, identifying the risk from each increment, the possibility of measuring the progress and the possibility of identifying the risks form each compilation process. The drawback of the iterative approach reflects on the management process. A software development project requires extra attention over the management process, the developer will not always fallow the schedule developed in the early development stage of the process raising the possibility of missing some of the objectives and using project development time for meaningless feature implementation.

As stated earlier in the report the project had a late start and there are only a few months for the submission deadline. The initial submission date was given as an approximation and the Gantt Chart and the schedule have been set for a period all the way to the 24th of May. The actual submission date is on the 10th of May, two weeks earlier and things will need to speed up.

The schedule in the Gantt Chart has not been followed accordingly, due to the actual submission date some of the early objectives have been willingly omitted or have been pushed in completion to other objectives. For example, the functionality of the game engine has been observed through building the development environment and much of the documentation of the Unity Jobs System. The improvement of the Edge-Detection method has been developed while programming the A\* Pathfinder method. C# coding skills have been acquired trying possibilities for AI prototype programming and so on.

There are still a number of objectives to be achieved in the project schedule, but so far even with the actual time limitation there is a high possibility of finishing the project on time. The only impediment is time, but I have managed to get free time from work, time which will be allocated for project completion.

# 6. Stakeholder

Regular weekly meetings are conducted with the stakeholder. Each week approximately five hours are spent examining the progress made to date, features that have been implemented, simple implementations that can be added to the project without affecting the development and submission date and the finally, results documentation.

Although the requirements of the project have been set in the early stages, with the development of the project, small modifications are added to the game, new actions are industrialised, new game frames, plans and characters are discussed and plans for additional projects are sketched.

The actual game development is being undertaken for quite some time and the reason was not very clear until recent meeting when the stakeholder stated that his game will not be developed to be just another commercial game. It follows a specific 3D zombie shooter script that asks for specific game play and game frames. With all the technology at hand in the present day, the only thing needed is the idea, and the idea must follow a storyline that will keep game enthusiasts in their seats, watching, while someone plays the game; then it’s a great game. For this reason, a small period of time in our meeting is allocated to discuss, promote or reject future development ideas.

The stakeholder has seen and is pleased with the development, as said earlier, not all the aims must be completed 100%, but as stated by him, it can always be room for more.

# 7. Progress

Progress has been made on the project. The stakeholder was pleasantly surprised about the values achieved and stated earlier in the report where major progress developments have been discussed. Although the project had a late start it is right to say that it is running well. There are still a couple of milestones to achieve but there is some time left to be allocated to development.

The next step in development is to remove individual threads from the main thread. This is a major step in C# development and if it is successful it is going to be a huge achievement. It can impact the order of the operations undertaken by the CPU and can be highly advantageous for the rendering process, but there is still a long way to achieve this progress.

# 8. Accomplishments

All the milestones achieved in the development process represent an accomplishment. The milestones are displayed in the Gantt Chart and they stand as a development schedule.

The initial Gantt Chart contains 18 milestones and 8 milestones have been completed, each milestone standing to a project objective. From the initial Gantt Chart the following milestones have been achieved:

Dynamic Cover-Based System for Unity Game AI Players:

1. Develop Unity Game Engine Skills;

2. Develop C# Programming Skills;

3. Familiarise with Visual Studio Platform;

4. Develop Coding Skills;

5. Set Up a Development Environment;

6. Enhance A\* Pathfinder Method;

7. Improve Edge-Detection Method;

8. Optimize Field of View Method;

# 9. Issues

The issues present in the development are the continuous feature updates of the game engine, features that cannot be added in the development of the game *Patient Zero* without refactoring a large number of redundant methods. The game is developed in Unity Game Engine version 4 which is focused extremely on single threading, meaning that all the actions must begin in a single place and return to update after completion. The new release, Unity 5, implements the concept of multithreading and a better allocation of processor cores for different operations, such as physics and rendering. Unfortunately, there will be an extensive work to move the game development from Unity 4 to Unity 5 or to add new component releases from the newly released Unity 5 into *Patient Zero.*

Another issue is the submission date, initially thought to be at the end of May 2019, which has moved two weeks earlier at the beginning of the month. Not having the exact submission details from the beginning of the project does add extra pressure over the process.

Continuous work is being carried out to combat the issues and to successfully complete and submit the project on the established deadline.