**University of Ulster**

**BSc (Hons) Computing Systems**

**COM563 – Organizational Process Focus**

**Improving game software rendering performance using CMMI**

**(Requirements Development Stage)**

**Student Name: \*\*\*\*\***

**Student Number: \*\*\*\*\***

Table of Contents

[Q.1 Requirements Development (RD) 3](#_Toc530875304)

[Introduction 3](#_Toc530875305)

[SG. 1 Customer Requirements Development 3](#_Toc530875306)

[SP 1.1. Elicit Needs 3](#_Toc530875307)

[SP 1.2. Transform Stakeholder Needs into Customer Requirements 4](#_Toc530875308)

[SG. 2 Develop Product Requirements 4](#_Toc530875309)

[SP 2.1. Establish and Maintain Product and Component Requirements 4](#_Toc530875310)

[SP 2.2. Allocate Product Component Requirements 5](#_Toc530875311)

[SP 2.3. Identify Interface Requirements 5](#_Toc530875312)

[SG. 3 Analise and Validate Requirements 6](#_Toc530875313)

[SP 3.1. Establish and Maintain Operational Concepts and Scenarios 6](#_Toc530875314)

[SP 3.2. Establish and Maintain a Definition of Required Functionality and Quality Attributes 6](#_Toc530875315)

[SP 3.3 Analyse Requirements 6](#_Toc530875316)

[SP 3.4. Analise Requirements to Achieve Balance 7](#_Toc530875317)

[SP 3.5. Validate Requirements 7](#_Toc530875318)

[Q.2 Metric 7](#_Toc530875319)

# Q.1 Requirements Development (RD)

## Introduction

I am currently working on a project designed to improve rendering capabilities in a game development software. The RD process outlines the main aims in the development process using measures from the CMMI-Dev v1.3 model.

The technological development in the past years led to major improvements in the 3D game development area. 3D software development is improving at a high rate, from a visual point of view a game frame is very close to reality in terms of rendering but, as the complexity of a 3D software grows the hardware improvement requests grow as well. This calls for programming development enhancements to allow the best use of the CPU’s capabilities in a multi-core system.

Main aim of the project is to improve the users game experience in a software and create, implement different prototype methods to improve the functionality and resource allocation in a software. There is a need to temper with the multithreading process in a software and create ways of running multiple tasks concurrently and detached from the main thread. All the threads start in the main thread, which dictates the completion order of different processes. Other methods are to be improved to sustain the minimum use of CPU features to improve functionality.

## SG. 1 Customer Requirements Development

### SP 1.1. Elicit Needs

For my final year project, I have chosen Unity Software Development. *Patient Zero*, a 3rd person shooter created in Unity has been set as the development environment, populated with assets, components and game objects developed in Unity. The aim of the project is to improve the rendering features in a software by tempering with the multithreading process and basic programmable game object methods such as A\* Pathfinder, Field of View and Ray-cast Edge Detection.

Ahead of the beginning of the project, I had a meeting with the project supervisor to learn about the project requirements and what the project involves. In C# development platform, the multithreading process starts in the main thread and once a thread completes a process it must synchronise with the main thread. One objective of the project is to develop prototype programming methods to allow the implementation of individual thread processes separate from the main thread, therefore taking full advantage of the processor architecture and enable different communication paths for individual threads. Another objective is to improve task completion time in a multicore architectural computing system where the rendering process is affected by the complexity of the basic programming methods implemented in the game frame such as A\* Pathfinder. All the current methods implemented are basic and preinstalled in Unity components.

In the meeting the main aims of the project have been sketched:

* Review the latest multithreading processes in place;
* Temper with the multithreading process;
* Improve A\* Pathfinder method
* Improve Ray-cast Edge Detection method;
* Improve Field of View Method

Project constrains, and expectations have been discussed. The main constrain of the project is time, but the supervisor stated that not all aims must be completed, even the completion of half the aims would mean significant improvements in game development software where any improvement is welcomed. The multithreading processes alteration has been discussed, but due to software limitations an alternative way has been adopted with the use of pointer programming methods which could allow the start of threads apart from the main thread.

The completion of the project would produce:

* Separate individual thread processes;
* Upgraded programmable game object methods;
* Improved software rendering;
* Possibility of using highly complex game software on devices with limited capabilities;

At the end of the meeting we conducted a search over the internet to locate forums and blogs focused on game software development, where issues arising from development could be discussed and resolved with help from other developers. Meetings have been scheduled periodically to asses project development and discuss and resolve any problems encountered.

## SP 1.2. Transform Stakeholder Needs into Customer Requirements

The customer requirements agreed following the meeting are as follow:

1. Evaluate the multithreading processing system;

Determine the functionality and complexity of the multithreading system in a single and multi-core processor architecture. Investigate any existing systems developed to improve the multithreading process and for any existing systems, evaluate functionality, implementation and establish advantages and disadvantages;

1. Temper with the multithreading process;

Develop prototype programming methods to remove individual threads from the main thread and create processes to sustain thread-switching hardware in game development software. The multithreading process is best for development environments where multiple threads run for a long period of time, however, having multiple small threads running for short period of times will tax the CPU’s resources and affect the rendering capabilities of the software.

1. Upgrade A\* Pathfinder method to run in real-time with the game asset grid attached to the AI object. The game object’s position has to be in the centre of the grid which moves instant with the AI’s movements. The grid’s game frame calculates the path and enables dynamic obstacle detection taxing the graphics rendering system as Field of View method and A\* Pathfinder method are using the same thread. The methods developed must have a low impact on the rendering system.
2. Upgrade Ray-cast Edge detection method implemented with obstacle detection purpose. Develop methods to reduce the calculation time of the grid and remove the Ray- cast method from the shared thread.
3. Upgrade Field of View method as a simulation of sensory implementation, every frame must run in real time to establish dynamic target detection. The Field of View method is the trigger method for A\* Pathfinder and Ray-cast edge detection method.

This stage of the project allows to control which requirement has more importance and needs more development time allocated.

## SG. 2 Develop Product Requirements

### SP 2.1. Establish and Maintain Product and Component Requirements

From the requirements listed above, sub-requirements have been developed and the requirements are broken down in smaller development stages:

1. Evaluation of the multithreading system;
   1. Breakdown the multithreading process in smaller tasks and evaluate;
   2. Understand the multithreading process in game development;
   3. Evaluate the newly implemented Unity Jobs System in Unity development;
   4. Research any existing attempts of tempering with the multithreading process, if any existing, evaluate and if possible, allow improvement in the following stages;
2. Multithreading process alteration;
   1. Create thread methods outside the bounds of the main thread and test results;
   2. Remove major thread processes from the main thread to test functionality;
   3. Implement pointer programming in the software and evaluate the results;
   4. Implement Unity Jobs System in the software and evaluate if any improvements of the rendering process;
3. Upgrade A\* Pathfinder method;
   1. Remove any unnecessary assets, colliders from the AI’s game object scripts;
   2. Reduce the amount of real-time calculations made in the AI’s script to generate the grid view;
   3. Increase the size of the grid cells from 0.5 to 1 to improve game performance;
4. Upgrade Ray-cast Edge detection;
   1. Redesign the standard Array iteration method implemented by Unity.
   2. Reduce the number of ray-casts deployed by the AI to detect game objects;
5. Upgrade Field of View method;
   1. Remove standard Field of View method from the game software;
   2. Develop Field of View method as an extension of A\* Pathfinder and Ray-Cast Edge detection method;

Processes completion allows further requirements enhancement for development while keeping the project in time constraints.

At the beginning of the project, the architecture, design and story line of the game have been set. As part of the development process, a game frame will be created with assets, game components and game objects from *Patient Zero* to be used as a development environment. The development of the game, *Patient Zero,* is at a stage where tempering with the main thread can lead to the end of the project.

### SP 2.2. Allocate Product Component Requirements

All the requirements have been specified in a formulated project plan. Having a project plan in place containing the requirements, I was able to set deadlines for different project areas and track the progress for each of them. I created a resource requirements list which states what resources are needed to develop the project and the skills level required to complete the process.

Further meetings between myself and the supervisor are set to further enhance the requirements development process.

### SP 2.3. Identify Interface Requirements

The project is set to be developed in a game development environment, meaning the product interfaces have been set at the beginning of the project.

There are three development interfaces for the project, each with their own requirements:

1. Unity Game Engine
   1. Provide a development environment;
   2. Drag and drop functionality;
   3. Game assets;
   4. Game components;
2. C# Programming Language
   1. Script Programming;
   2. Intended for Rapid Application Development;
   3. Object Oriented;
   4. Provide access to Microsoft .NET Platform;
3. Visual Studio
   1. Build .NET Core applications;
   2. Test, debug and deploy;
   3. Build 3D games in C#;
   4. High time to launch application.

This process will be reviewed at a later stage, due to project development more interfaces may be created. Setting the interfaces for the project confirms that the requirements of the project can be achieved.

## SG. 3 Analise and Validate Requirements

### SP 3.1. Establish and Maintain Operational Concepts and Scenarios

The requirements, together with the improvements brought to the functionality of a software, have been presented as a general description at the start of the project. The project is in continuous development with the refinements of the requirements.

For development purpose a game frame has been created to apply and test the developed methods onto an environment and evaluate the outcome, interaction and functionality. The development environment and the operational concept will be used together in project development.

### SP 3.2. Establish and Maintain a Definition of Required Functionality and Quality Attributes

At the start of the project the supervisor presented the expectations of the project and the outcomes of a successful development. The requirements of the project have been reformed, sub-requirements have been created and prioritised to ensure a proper time development management considering the time constrains.

### SP 3.3 Analyse Requirements

During project development, basic implemented game object methods will be improved developing enhanced sensory process implementations for AI game objects. The successful improvement of the multithreading process will enhance and accelerate the rendering process for a better game experience. The methods developed could be used in other game software to improve functionality without consuming time to develop methods that are already in place.

A record will be kept during the development process to measure the technical performance of the software during the project development time.

### SP 3.4. Analise Requirements to Achieve Balance

To analyse the requirements of the project and the constrains, I have discussed the risks of every requirement set for the project with the supervisor. The discussion revealed the requirements with high development risk and time to completion constrain. One of the requirements, tempering with the multithreading process, is highly time-consuming and we decided that smaller time development intervals are to be set for tasks completion in this requirement. If no successful implementation has been made in the designated task time, the development will continue with the next task. The alteration method to start with will be to use pointer programming methods to alter the multithreading process. If the development time set for the project will allow, all the tasks that are involved in the multithreading process which are not successfully completed, will be revised and another completion attempt will be made.

### SP 3.5. Validate Requirements

In the development process, the supervisor and I settled that all the requirements developed for the project completion will have a successful outcome. Every requirement has been discussed and risks have been identified. Prototype programming methods will be developed, tested and the outcome will be subject for feedback and any changes that are required. This stage will provide requirements that have not been listed and will ensure that the final result of the project will function as intended.

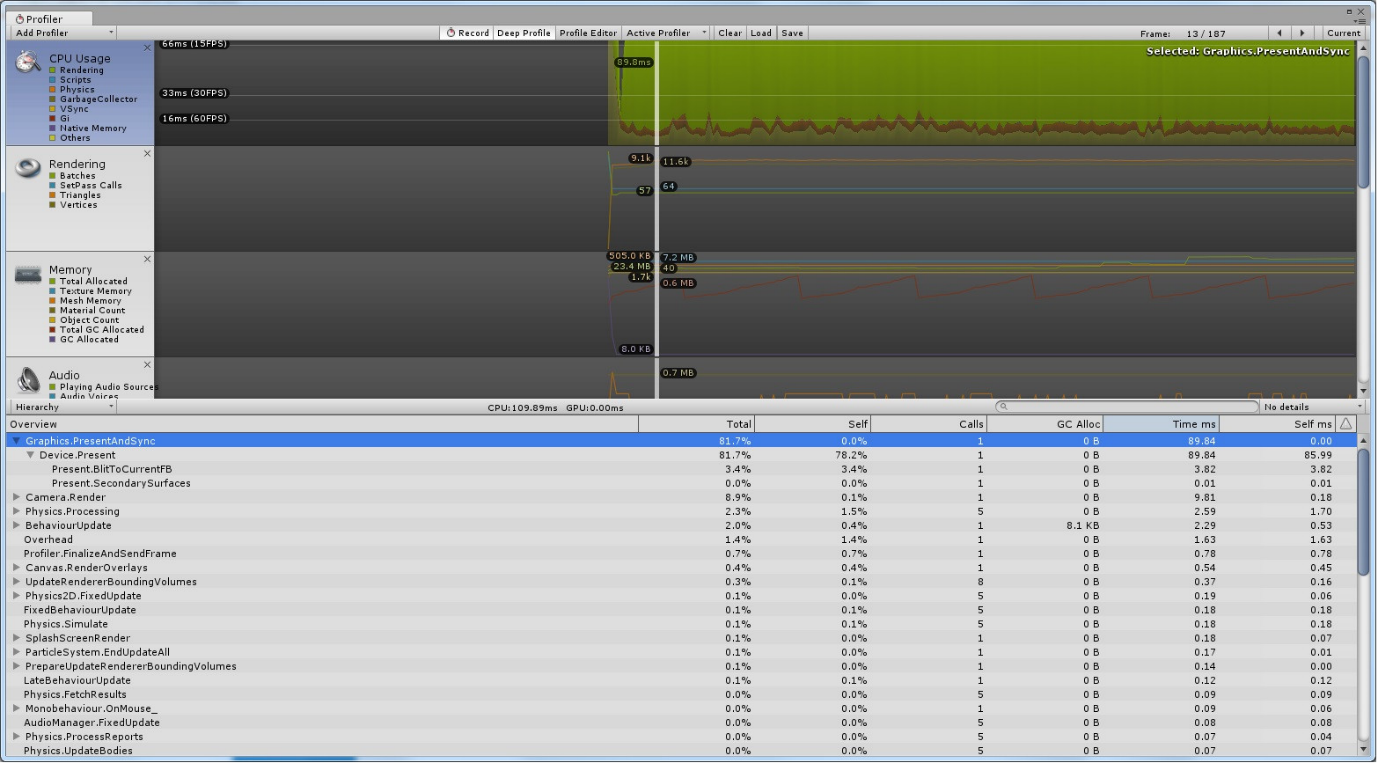
# Q.2 Metric

The aim of the project is to develop programming methods to improve the functionality and resource allocation in a software and to improve the user’s game experience. Another aim is set to develop prototype programming methods designed to temper with the foundation of multithreading and create processes to sustain thread-switching hardware in game development software. The above stated aims stand to improve the rendering capabilities of a software.

To measure the progress made during the development process, a Unity game engine tool will be used, called Unity Profiler. Unity Profiler helps optimise a game software by broadcasting the time spent in different processes of the software. Offers important feedback about the rendering process, game logic and animation. It will record software data performance, present observations in data improvement or regression following the methods implemented. Unity Profiler will help control and view software optimisation.

Unity Profiler can measure all the processes running in Unity such as:

1. CPU Usage
   1. Rendering;
   2. Scripts;
   3. Physics;
   4. Gi;
2. Rendering
   1. Batches
   2. Set Pass Calls;
   3. Triangles;
   4. Vertices;
3. Memory
   1. Total allocated;
   2. Mesh Memory;
   3. Material Count;
   4. Object Count;
   5. Total GC Allocated.



The project’s goal is to improve the rendering process. Along with the rendering process development the scripting process will be improved. At the development point of the project environment, the game frame with the necessary assets, the rendering process has been measured at run time. The rendering measurement set as a project baseline is 30FPS (frame/second), anything lower and functional can be counted as a success. The same applies for the scripting process where the first measurement has been set at 15FPS.

When receiving data, the questions standing will be answered with sub-questions: Why? Where? Cause? Can it be fixed? A measurement plan will be set to gather and analyse measured data. An important factor to be taken in consideration beside the development environment, which has to remain standard throughout the whole development process, is that the measurement must be taken on the same device on which the first measurement was taken. Otherwise the data gathered from the measurement process will not be accurate.