Method One – Size Maps

Both of the size maps, disk memory and RAM memory do not show anything worth optimizing, as the biggest numbers belong to already well optimized template from UE5 and anything that I added is not beyond reasonable memory usage. This method could have shown me high RAM usage, where lower end computers would have had a problem with, fortunately the project is too small to have this issue.

Graphical user interface

Description automatically generatedGraphical user interface

Description automatically generatedMethod Two – Light Complexity

At the momeny both my AI and player models use point lights attached to them, as part of making the scene look futuristic and making the enemy slighly easier to spot in the distance.

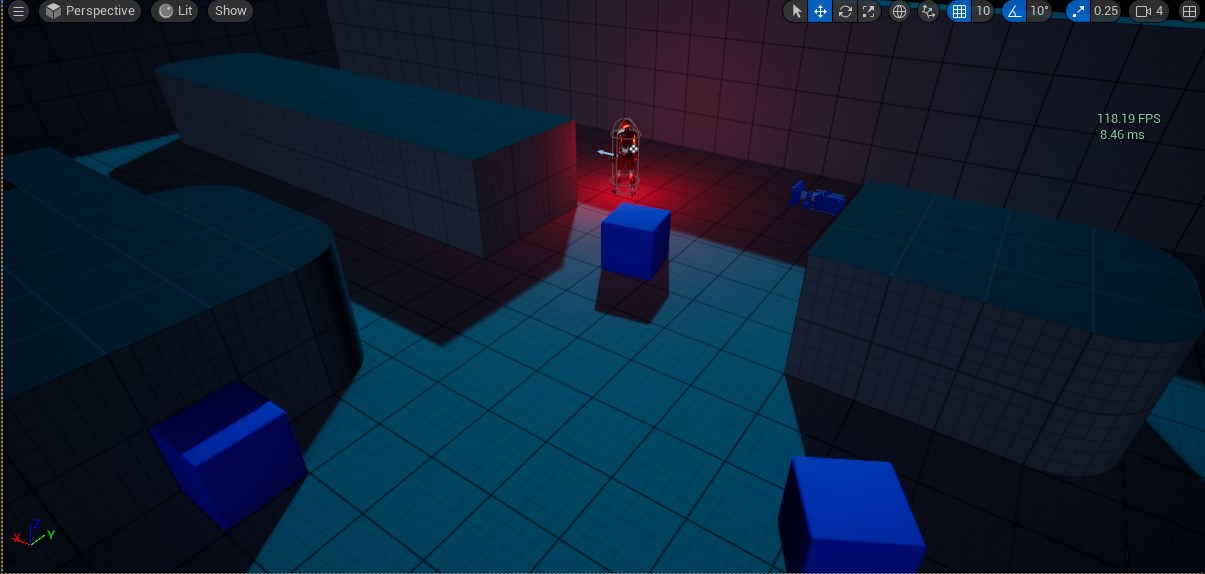
A picture containing background pattern

Description automatically generated

This being light complexity with just hostile.

Background pattern

Description automatically generated

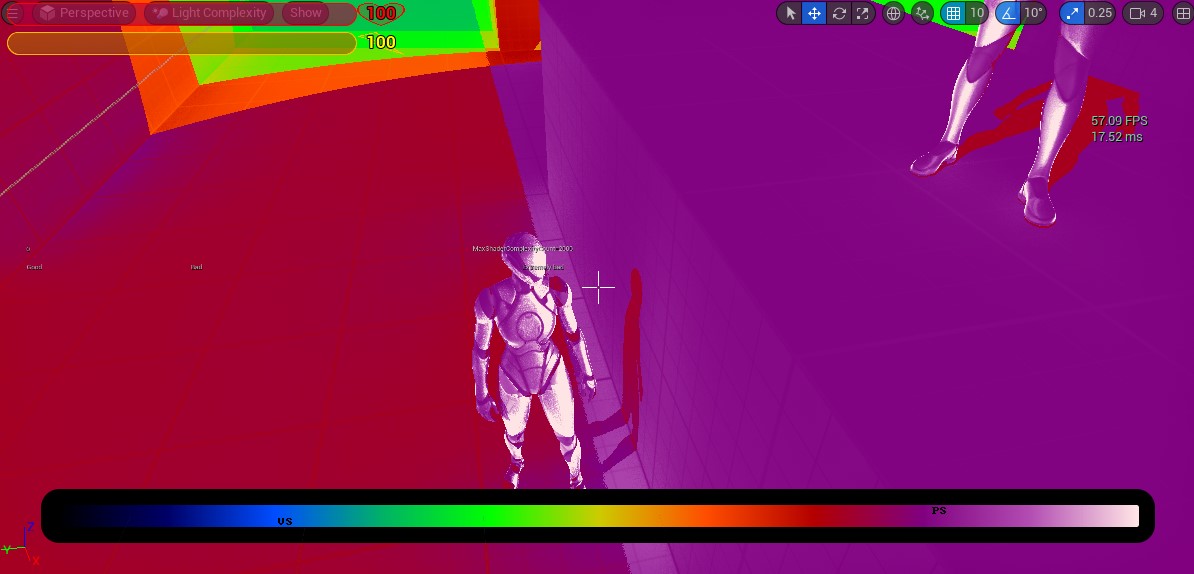
This being light complexity with both player and hostile AI at the same time, as you can see it’s using much more resources to render this area due to the light being emitted by both models, this is still not a massive usage for the current size of the projects and most computers would not struggle with this, however if in the future there would be more than just 2 models being used with those point lights the problem will become quite big. So let’s get to optimizing the lights without removing the effect from them. 

[This being the cool effect of dark map, and point light]

Optimizing the light

Taking a closer look at the point light I decided on two changes to improve the FPS. Light intensity was decreased by ¼ meaning that rendering distance of the light and area in which it affects environment is now smaller, with both player and AI being at the same location is was exivelent to not spawning ½ of one light.

The second change was to disable cast shadows option for both point lights which stopped the render of one of most resource heavy operations. The lack of shadows coming from both points lights managed to boost performace in the area of two models(or two point lights) by about 33%.



FPS in area of two lights with shadows and high light intensity is about 60FPS on average(57.09 on screenshot).

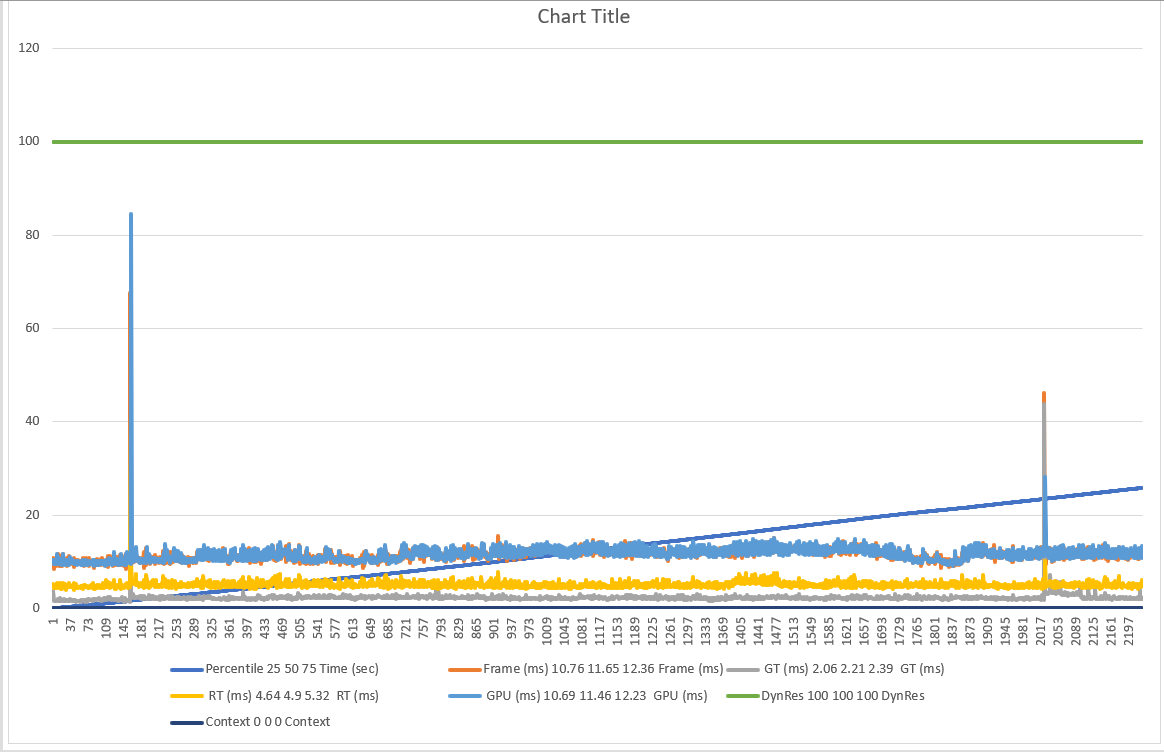
A screenshot of a video game

Description automatically generated with medium confidence

FPS in area of two lights without shadows and lowered light intensity is about 110FPS(109.52 on screenshot)

Method Three – Process Chart in Build

Graphical user interface, text, application, email

Description automatically generatedThe chart is showing two spike sin GPU render time, the first spike is at the moment of loading the level, while 2nd spike is at the time of loading main menu after pause.



There is also a moment of pausing the game visible on the chart.

Graphical user interface, text, application, email

Description automatically generated



At the same time it’s also visible when enemy got close to the player, which is the moment where two points lights were closest to each other causing most intensity. As I was playing and recording a video of the game so I can match events on graph to events in the game, the enemy run into the player twice and from the video I can mark the point on the graph.

The more interesting thig is that the first contact caused more time needed to render the frame compared to the second time.