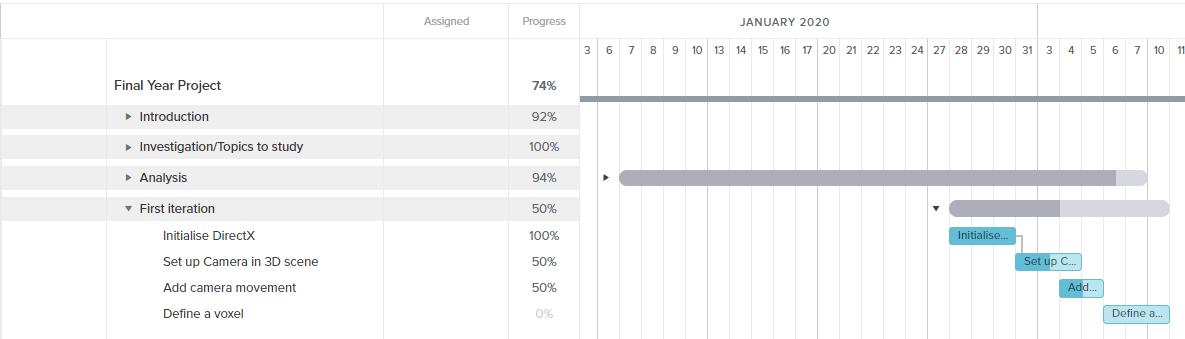
# Iteration 1: 27/1/2020 – 10/2/2020

## Gantt Chart



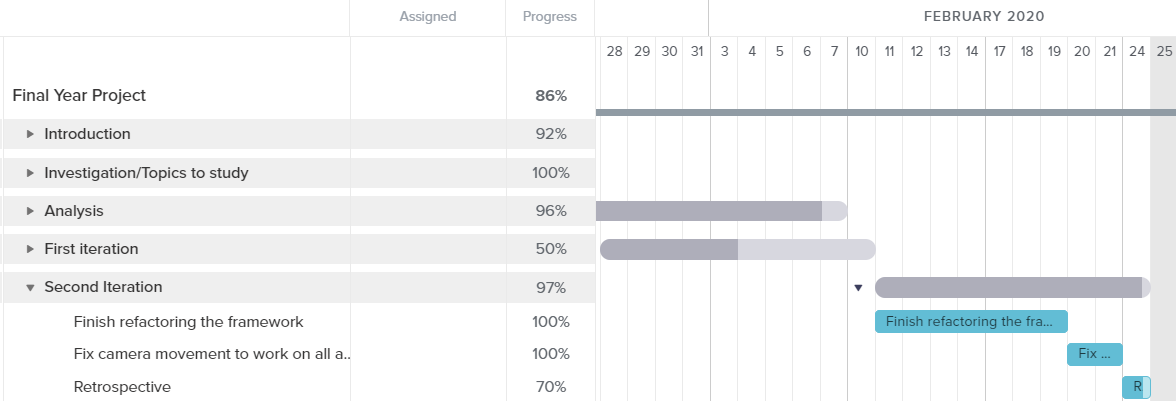
## Review

The first iteration was about selecting an API, producing a product backlog and prioritising the desirable features using the MoSCoW method. In addition to these tasks, further time was spent on studying the provided framework and ensuring that the initialisation of Direct3D and basic camera set up were in place. As you can see on the Gantt chart above, these tasks were set in parallel with the last part of research analysis and I allocated more tasks that I could achieve thus pushing some further functionality such as camera movement and defining a voxel to the following iterations.

## Retrospective

On reflection, I gave too much time to myself for certain non-programming tasks but on the contrary when this was realised, I gave much more tasks to complete until the end of the iteration. I will take that into consideration and allocating tasks during the next sprint plan.

# Iteration 2: 11/2/2020 – 24/2/2020



# Review

In the second iteration, I finished refactoring the framework. I separated Direct3D related functionality into its own class called “Graphics” and everything else that relates to the actual implementation in the “Application” class. A recap on initialisation of direct3D was needed for debugging issues occurred during refactoring. A camera was created and separated into its own class to compute all the needed matrices in one place. In Addition, a directed input class replaced Win32API to get input from the user. Win32API was designed for applications where users enter data through keyboard and not for application where quick real-time input would be ideal. There is a lot of extra processing when getting input from Win32API as it needs to convert each key to ascii while also there are some extra steps for windows special keys such as “alt” which is redundant to games as a frame cannot be rendered until the message queue has emptied. As a result, a “free-look” camera movement was implemented by receiving direct input from mouse and keyboard.

# Retrospective

Up to this point, the sprint goals have been mostly met as the API has been chosen and the basic framework of the application has been refactored. It could have been quicker if knowledge of initialisation of direct3D was thoroughly understood from the start as the debugging process would speed up. Better time management should be taken into practice as more time is consumed on writing and coding and less on planning ahead which if done too late, I can be easily caught up with irrelevant tasks and lost track of my goals. On the other hand, having a framework that I understand deeply helps on debugging issues that may occur in the future in regards of voxel rendering.

# Iteration 3: 11/2/2020 – 24/2/2020

# Review

# Retrospective

What did you accomplish?

Did you meet your sprint goal?

What worked really well this sprint? What could have worked better?

Are there any productivity blocks that kept cropping up? Review all your Daily Scrum videos for this information.

Is there anything that needlessly stressed you out, or that you really enjoyed?