FYP Interim Report 2023-2024  (2 page limit)   
A4 12 point single spaced Times font (or equivalent).   
Name: Jevgenij Ivanov  
Student Number: 20748055  
Project Title: Room Generator Extension for Godot 4.1  
Supervisor: Ralf Bierig  
ECTS Credits: 15  
  
1) Overall Project Objectives.

The primary objective of Room Generator Extension project is to streamline the process of creating 3D environments within the Godot 4.1 engine, enhancing the speed and efficiency of building such spaces from scratch.

By developing an intuitive and user-friendly plugin, the aim is to empower designers and developers to quickly assemble scenes using a library of prefabricated rooms and various components.

The plugin will be installed effortlessly by adding the plugin directory to “addons” folder in any project. This addon is then activated within the project settings, and users will immediately see a new tab appear in the bottom right section of the Godot editor interface, with all the tools available to be used.

This tool will not only be used to accelerate the manual creation of expansive and complex structures but also to ensure that these constructions adhere to high-level design principles. These principles are planned to include features such as avoidance of redundant elements, for example double walls or overlaps that can potentially hinder the aesthetics and performance in games and other 3D projects.

2) Description of work completed.  
  
2.1) Evidence of work completed. (delivered outputs, number of lines of code, Class Diagram or similar)   
  
3) Outline of future work. NOTE: Additional Evidence Appendix - You may include additional “Evidence of Work Completed”, (e.g. listings, literature references, figures and screenshots) as an Appendix – this does not count to the page limit.

As an aspiring video game developer, I will do my best to create a plugin that I myself will want to use to create my future games, this way I also believe that I can create a something that other people can use and not just me.

One of the first major issues I encountered during the plugin development, was that the CTRL+Z and CTRL+Y commands (undo, redo) did not work as intended. One of the bugs, for example, when adding a box through the plugin, and then pressing CTRL+Z to undo the box, Godot instead would undo the last action you did in the Godot editor, prior to any actions in plugin. This was quite an irritating behaviour, so it was the first issue I decided to tackle and develop the correct behaviour between plugin and Godot editor.  
  
Previously, I learned how to implement Memento and Command patterns to achieve undo/redo functionality in my C# projects. These complex design patterns were not necessary for JDScript code in my plugin script, and instead, I could simply use Godot’s “get\_undo\_redo()” object, and then utilize the objects methods, such as “.create\_action”.