**Spike:** Task 12

**Title:** Command Pattern

**Author:** Luke Valentino, 103024456

**https://github.com/LukeValentino138/COS30031-2023-103024456**

**Goals / deliverables:**

A robust command parser and manager for the Zorkish game. Including commands such as GO, HELP, INVENTORY, LOOK, ALIAS, and DEBUG TREE.

* Code see /task12/
* UML Diagram

**Technologies, Tools, and Resources used:**

List of information needed by someone trying to reproduce this work

* Visual Studio
* https://cplusplus.com/reference/
* Swinburne Games Programming Lectures

**Tasks undertaken:**

* Add items/entities to the map data and logic.
* Develop command processor logic based on necessary functionalities. (e.g., GO, HELP, INVENTORY, LOOK, ALIAS, and DEBUG TREE).
* Develop command abstract class to be inherited by every command class. This needs to include a constructor, destructor and execute method.
* Make each command class according to the specifications. These classes should inherit from the command base class.
* Create a commandManager class. This class needs to process the user input and call the appropriate command execute.

**What we found out:**

Separating the command logic into multiple subclasses and using correct OOP principles allows for significantly more modularity and ease of implementing new commands. Using a dedicated command manager also helps achieve this.

UML: