COS 214 Project - The magnisifint 7 - Task 1:

1.1) Identify the functional requirements.

The functional requirements are as follows:

Each engine needs to be accessible and created individually from each other in the same fashion that they would be created in a real rocket design scenario, not just cloned. They need to be able to be individually iterated through and tested to ensure that they are all functional before the actual launch takes place. This is what will be known as the static test. This will also need to ensure that the rocket can communicate with each engine.

Once the static test is complete, the rocket will need to be updated to show this development and that we can go ahead with the actual launch of the rocket.

There will need to be a differentiation between the stage 1 and 2 engines (Merlin and vacuum merlin) used as well as the differentiation between the different rocket types (Falcon-9 and Falcon Heavy)

The spacecraft will have two parts. Stage 1 and Stage 2. Stage 1 will need to get the spacecraft from the ground and almost into orbit and will require the most engines and fuel. Stage 2 will provide the final kick to get the payload into orbit.

The different payloads are as follows:

Crew Dragon

Dragon

Starlink Satellites

Based on the modular nature of the space. Each module will need to work with the other modules and each combination possible. (Save for Starlink as that will always be attached to a Falcon-9).

The starlink payload will need to hold up to 60 satellites and release them when instructed.

The spacecraft will need to be able to communicate with the base once the launch has begun and be controlled from the base.

The spacecraft will also require the functionality to be stored before a simulation takes place.

Simulations of the spacecraft’s launches will need to be made with the functionality of being able to pause the simulation at different stages, being reset and continue the simulation after adjustments have been made. This will be known as the Test Mode Simulations.

There will also be simulations of the actual launch where it cannot be interrupted known as Actual Launch Simulations.

1.2) Design the processes using Activity diagrams.

Files: “/Initial Activity Diagrams/”

1.3) Decide on the patterns to address the functionality defined by the functional requirements and processes.

Memento, State, Composite, Command, Builder, State, Adapter, Prototype, Factory Method, Abstract Factory, Iterator and Template Method.

1.4) Design the classes for each of the identified patterns taking their interrelationships into account.

Files: “/Initial Classes and Class Diagrams/Concepts/”

1.5) Draw a class diagram of your system.

Files: “/Initial Classes and Class Diagrams/Class Diagram Filnal Concept V01.pdf”