Solar System Challenge

Mariana-Ionela Muntian

Overview

The Solar System Challenge was implemented using:

- **Java** for backend logic and computations.
- **JavaFX and Scene Builder** for the graphical user interface.

To enhance development efficiency, OpenAI was utilized as an assistant. However, the **solution's structure, organization, and most of the code** were developed by the author.

How to Run the Application

The application can be launched by running the `HelloApplication` class. The interface serves as the primary mode of interaction, as **user input is only accepted through the interface**, while outputs are displayed both in the interface and the console.

Usage Recommendations

Due to time constraints, the application is not fully refined. For optimal functionality, it is recommended to first press the button corresponding to **Stage One** before proceeding to later stages.

Structure

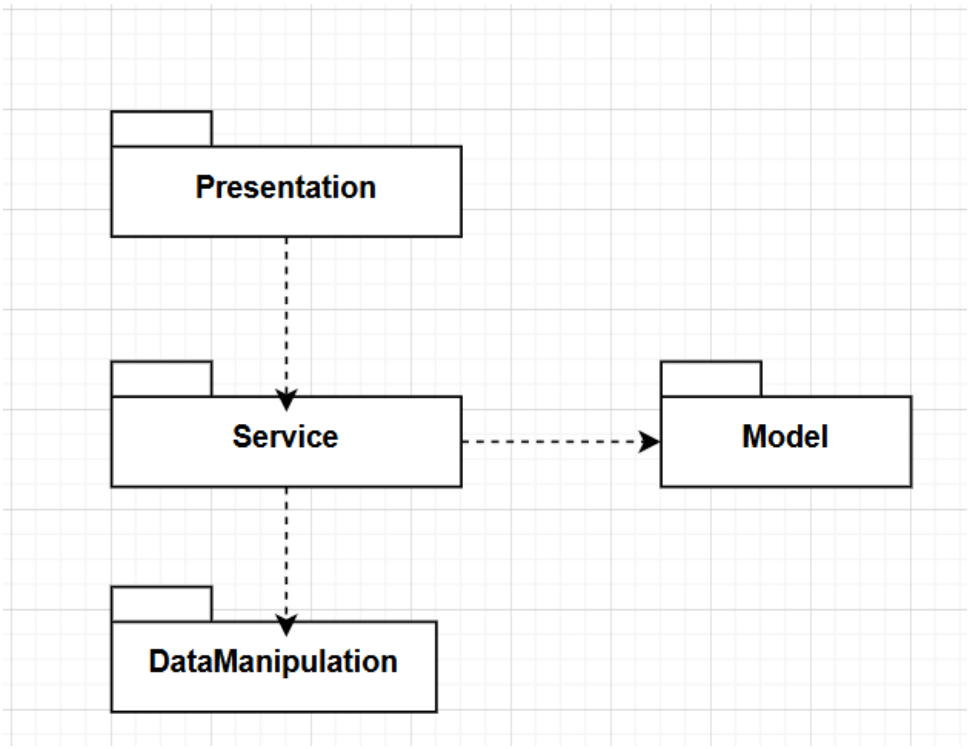

Figure 1: System Architecture

A. Presentation Package

Contains the classes that control the flow of the application and interact with the GUI:

- `HelloApplication`
- `HelloController`
- `MainController`
- `StageOneController`
- `StageTwoController`
- `StageThreeController`
- `StageFourController`
- `StageFiveController`
- `StageSixController`
- `EndPageController`

B. DataManipulation Package

This package is responsible for extracting and parsing data from input files. It contains:

- `ReadData`

C. Service Package

This package performs the actual computations needed to solve the challenge:

- `ComputationsStageOne`
- `ComputationsStageTwo`
- `ComputationsStageThree`
- `ComputationsStageFour`
- `ComputationsStageFive`
- `ComputationsStageSix`

D. Model Package

Contains the core classes representing the objects in the simulation:

- `Planet`
- `Rocket`
- `SolarSystem`