Seng201 Project

Student Name: Nakul Reddy **Student Name:** Shivin Gaba

Student ID: 41548232 **Student ID:** 38266964

Application Structure:

While planning the game we observed that lot of classes would have similar attributes and characteristics, so it made sense for us to use Inheritance in this project. Inheritance was the main recurring theme in this project and was one of the fundamental programming features project was built on.

Items

The class Items is a superclass that has four subclasses Medicine, FoodItems, Money and SpaceshipPart which inherit the qualities such as price and quantity. Medicine and FoodItems are superclasses to the different types of medicines and food items. They have their own attributes such as healing power and nutrition.

CrewMember

The CrewMember class is a superclass to the different CrewMember types which are Martian, StarLord, Nebula, Medic, Human, and Transformer. These sub classes contain their own special attributes such as special powers in this case. There were few specific methods in CrewMember class that were implemented differently in the subclasses, but the intended functionality of the methods was maintained.

Planet

The Planet class is a superclass to the different planets like Miller’s planet, Cooper’s planet, Lumbar planet, Osiris, Solaris, Edmund’s planet and a SpaceOutpost planet.

Collections

Collections have been used extensively in this project namely in the form of ArrayLists. ArrayList have been used to store FoodItems, CrewMembers and Medicines, as it made it easier for us to access the objects for each food item, crew member and medicine.

Medicines and FoodItems

In the case of Medicines and Food items they have been used as an inventory where the crew stores the different kind of FoodItems and Medicines that the Crew have purchased from the Space Outpost. The different members can then consume whatever the crew has purchased.

Crew Member

In the instance of CrewMembers the ArrayList has been used to store the CrewMembers that make up the crew. The ArrayList that corresponds to the CrewMembers has been particularly useful as it has also helped us to generate tabbedPanes for each CrewMember in the GUI setup.

Collections in GUI

In GUI ArrayList has been used for JLabels, JProgressBars and JRadioButtons. ArrayLists when used for JProgressBars, proved to be very convenient to update the health, hunger and tiredness levels of each crew member when a particular crew member was hit by the space plague or performed an action. JRadioButtons are used for the pilot functionality. ArrayList in particular have been very useful in disabling the crew member's radio button under pilot pame that died due to the space plague.

Planet

An ArrayList was also used to store different Planet objects. This array list was called in random events that assigns a random item to a random planet.

Unit Test coverage

The Junit test coverage came up to 19.2%. Separate JUnit test cases were made for Crew, Crewmember, FoodItems, GameEnvironment, RandomEvents, and SpaceShip.

The methods from the remaining classes have been indirectly tested using the existing Junit test cases. The GUI screen classes were manually tested and could not be tested with test cases. The methods that were tested in the JUnit test cases were those that involved logic such as if, else if and exceptions. Getter and setter methods were not tested directly but were tested indirectly as well. The RandomEvents class received very little test coverage since the methods in this class involved function calls and not just returning random numbers.This led to a relatively low unit test coverage. The searchPlanet and pilotShip method in the CrewMember class could not be tested completely since they were incorporated with the GUI and involved method calls from the RandomEvents class.

Feedback

We believe that the project has improved our knowledge of Object-Oriented Programming, increased our competence in programming in Java and working with GUIs. It helped us to understand inheritance, collections and testing. It helped improve our understanding of the theoretical side of the course in general.

The project in general went relatively well and was a success. The fact that we started the project during the term break gave us a good head start and we were able to get help from the tutors on fine and minute details that helped us to deliver the project in time. This also gave us enough time to work on our GUI and make it look presentable. Our lack of knowledge of using file sharing systems like Gitlab and Github was an issue and was something that we struggled with throughout the project.

Effort and Contribution

In general, the workload was balanced between the two of us. We spent around 8 hours every day for two weeks throughout the term break and three hours every day after the break. Shivin was mainly responsible for the GUI layout, the design, creation of the crew member characters, food types, medicine types, planets and sound. Nakul was mainly responsible for creating functionality for each character, medicine, food and linking the GUI to the backend.

Both of us contributed 50% in the completion of the project since we worked on it together all the time. Neither of us did project related work outside of when we met each other at university.