

## Island Trader Final Report

- **The structure of my application:**

I decided to use CSV files to simulate a database, I did not want to have to create a class to instantiate every object like Item or Upgrade. The Repository classes are responsible for reading the files and creating the objects. Every Object in the game but the Trader is created using a repository with a .csv file. In this way, all the logic for getting data from CSV (which looks like a database) is in the repository layer.

The islandtrader.core package has the core classes for the game, and the repository package will use those classes to instantiate the objects for the game. The service package is a helper for random events.

Then I used inheritance for the Items and Upgrades, as both are goods that will be traded, they are Tradable, and the StoreTradable is used when populating the Stores with the goods and upgrades with the correct price for that Store.

I also used inheritance for Pirate, UnfortunateWeather and RescuedSailor to extend the RandomEvent.

Creating an object from the CSV files and reading them in the pirateRepository for example, made it easy to be able to have different pirates or unfortunate weather situations or sailors to rescue.

They have a probability of encounter and are added randomly to the Routes when the Islands are created in the IslandRepository.

The IslandRepository is the more complex repository, as it is from there that almost all other repositories will be called to instantiate the objects. Item and Upgrades to be added to the Store, the store for the Island, and the Routes available for that Island. That made one method there bigger than I wished, but I could not go back to try to refactor it any further.

Four routes are leaving from each Island. Every time the game starts the random events change, each route has three random events that are possible to happen, with different probabilities of happening, and one could happen with no event or one of the three. These will change every time the game starts, for example, a different pirate in that route every time the game starts.

- **An explanation of unit test coverage, and why I managed to get a low percentage coverage.**

General test coverage: 9%

Repository package: 0% - It is not tested because it has mostly private methods, and the CSV files references are static so I would need other test.csv files instead of the real ones, but I did not get the time for that.

Service package: 96.8% - It only has one class to help the random events.

Islandtrader.core: 30.9% - I tried to focus on the most relevant methods and the public ones. I would need to make changes in the code to cover more, and as most of them are simple getters and setters with less relevance I chose to skip them.

I have attached the HTML test coverage report in the zip files.

- **My thoughts and feedback on the project.**

It is a nice idea for a game. I wish I could have done it properly with all the requirements and tests and the extra credit tasks. It gave me valuable experience on working on a project from start and using the GitLab (branches, Merge Requests, Code Review) and Jira Agile board, tools that are used in real-world IT companies.

- **A brief retrospective of what went well, what did not go well, and what improvements I could make for your next project.**

For me, it was a great opportunity to develop and manage a project.

Learning how to use LucidChart, GitLab and Jira Agile board was very helpful, especially for the future of my career, unfortunately for this project, it ended up being pointless as my partner could not bother to check the agile board or the UML diagram.

I am aware that the tests should be done as the code is being developed from the beginning, however as I started to worry that there would be no application at all to run in the demo, I decided to do it later, and my partner dropped so I am glad I did it, then at least I have an application to deliver, with most of the requirements.

For my next project, I have learned to be more assertive with workmates. I did not like or want to feel like I must beg my classmate to do his share, but next time I will be more direct with weekly deliveries and from the beginning inform and ask help from the teaching staff with the situation, hopefully, it will motivate my next partner into doing the work (and with quality). Or even better, make sure the partner I get is really interested in an A+ grade (as I am) and doing the work early, even ask for recommendations from other people, as we do before hiring someone. Turns out I prefer my grades to be as good as possible rather than making friends. And I cannot help but feel frustrated that It was a nice project, I like games, so I was looking forward to making this game and ended up with a project that is not meeting all the requirements and as such will reflect on my marks. It is being very stressful sacrificing other papers to still have an incomplete project to deliver.

- **The effort spent (in hours) on the project per student.**

~150 hrs

I did not write down exactly the amount, it started around 10 in the first week and since the partner dropped the paper, I have worked on the project what feels like “all the time”. I am going to round it up to 150 according to what I remember from the beginning of the project up to these last weeks.

- **A statement of agreed % contribution from both partners.**

Maria H Balzaretti: 100%