CS 2340 Spring 2020 – Milestone 5

Ships, Marketplaces, and Robustness Diagram

BACKGROUND: For this iteration of the project, your team will need to implement a ship for the Player, and a Region Characteristics UI, as well as some functionality to trade at a region. Robustness diagrams are used to visualize the connection of major components inside of your game. These diagrams can help you ensure that your system is adhering to a software architecture that reasonably separates responsibilities and visibility.

PURPOSE: Planning is key when designing large systems. This milestone expands upon the planning started by M3 and M4 through the creation of another diagram. In industry, diagrams like the robustness diagram are commonplace in the design phase. In fact, most work that occurs in creating software lies within the design phase.

TASK: This milestone has one team design deliverable, a required feature set for implementation, and a demo which will occur the week after milestones are due.

Robustness Diagram (Team Deliverable)

1. For the character creation we required you to implement in M3, generate a robustness diagram based upon this system.
    - Specifically, ensure that both the Welcome Screen, the Character Creation Screen, and the Character Sheet Screen are components of the diagram.
2. Your diagram must include any necessary controller classes needed to make the system operate.
3. Your diagram must include an entity class to store the Player data after creation and for retrieval.

Implementation Requirements

1. Design and implement the Region Characteristics Screen.
    - This screen should display information like the region name, tech level, etc.…
2. Design and implement a Ship for the Player
    - The Player has a ship with the following attributes:
        - Name, Cargo Capacity, Item Inventory, Fuel Capacity, Health
    - For this milestone, you only need to implement one specific ship.
3. Design and implement a Marketplace for each Region.
    - Each region must have its own marketplace, with partially unique prices and goods.
    - The goods available at each market must be determined by the tech level of the region that contains the market.
        - There is a chart below which gives an example set of goods and their corresponding tech level, along with any other information relevant to them.
    - The price of goods at each market must be determined by an algorithm which considers a few variables. You should create your own algorithm.

    - Example algo: base_price * (tech_level – minimum_tech) + random_variance = final_price
- The Player must be able to buy and sell goods at a market.
    - These goods must be added to the Player's Ship inventory.
    - The buying and selling prices should be different from each other.
    - Make sure that you aren't allowed to buy goods when you are at cargo capacity, and vice versa!
    - The Player should be able to buy and sell by clicking buttons on the Marketplace UI.
    - This UI should also display how many of a certain item the Player has on their Ship.
    - The Player's **Merchant Skill** should influence the price on goods.

Extra Credit

Character Upgrades: Add functionality to purchase character upgrades from the markets. You will also need to implement a UI to equip, remove, and view equipped character upgrades. An example might be a weapon that increases fighter skill or a scroll that increases merchant skill. This extra credit is intentionally very opened ended to allow you to showcase your creativity.

Checkstyle

During demo your team will be required to run the checkstyle script (located under files>checkstyle>Java Guide.pdf). This script will give your project a score out of 10 and will account for 10 points of your M5 final grade. Be sure to run the checkstyle script prior to submission to avoid unforeseen deductions.

Milestone Tagging

Tags are a way of marking a specific commit and are typically used to mark new versions of software. To do this, use "`git tag`" to list tags and "`git tag –a tag_name –m description`". You are required to tag the latest commit **<u>before the deadline</u>** which is to be graded during demo. You will be required to pull this commit during demo.

Submission Requirements

In addition to your diagrams, ensure that you include a link to your GitHub repository in your submission. Also, ensure that you have added your grading TA(s) as collaborators so that they may view your private repository. Repositories must be located on the Georgia Tech GitHub and must be set to private! Points may be deducted if these guidelines are not followed!

CRITERIA: You will be graded according to the rubrics on the final pages of this assignment document. Please note, 75 points of the 100 total points are dependent on your group's demo. **Groups are required to demo in order to receive credit for the features they have implemented. If you would like to demo changes made after the milestone due date, you will receive a flat 20-point penalty on the milestone's grade.**

Robustness Diagram Rubric (25 points)

The 3 M3 screens are on the diagram, formatted correctly.	10 points.
Player data included in some form as an entity class.	5 points.
Diagram contains any necessary controller classes to connect components logically.	5 points.
Connections are present between necessarily connected components.	5 points.

Implementation / Demo Rubric (75 points)

A link to the group's project GitHub repository is provided in the submission and all grading TAs have been added.	10 points.
Previous functionality is still functional.	10 points.
Region Characteristics Screen exists and provides relevant information.	5 points.
Ship class exists and contains the required fields.	10 points.
The Marketplace UI exists with a reasonable set of components (buy/sell/count/etc) and allows the Player to buy and sell goods.	10 points.
Each Region has a market which can be used.	5 points.
Each Market has a distinct set of items based upon the tech level of the region.	5 points.
The pricing of goods is calculated using some algorithm (not fixed pricing).	5 points.
The Player's Merchant Skill influences the pricing on goods.	5 points.
The Player cannot buy goods without enough money, buy goods without enough cargo space, or sell goods they do not own.	5 points.
The Marketplace UI shows the Player how many of each item they have on their ship.	5 points.
EC: Character Upgrades	10 points.