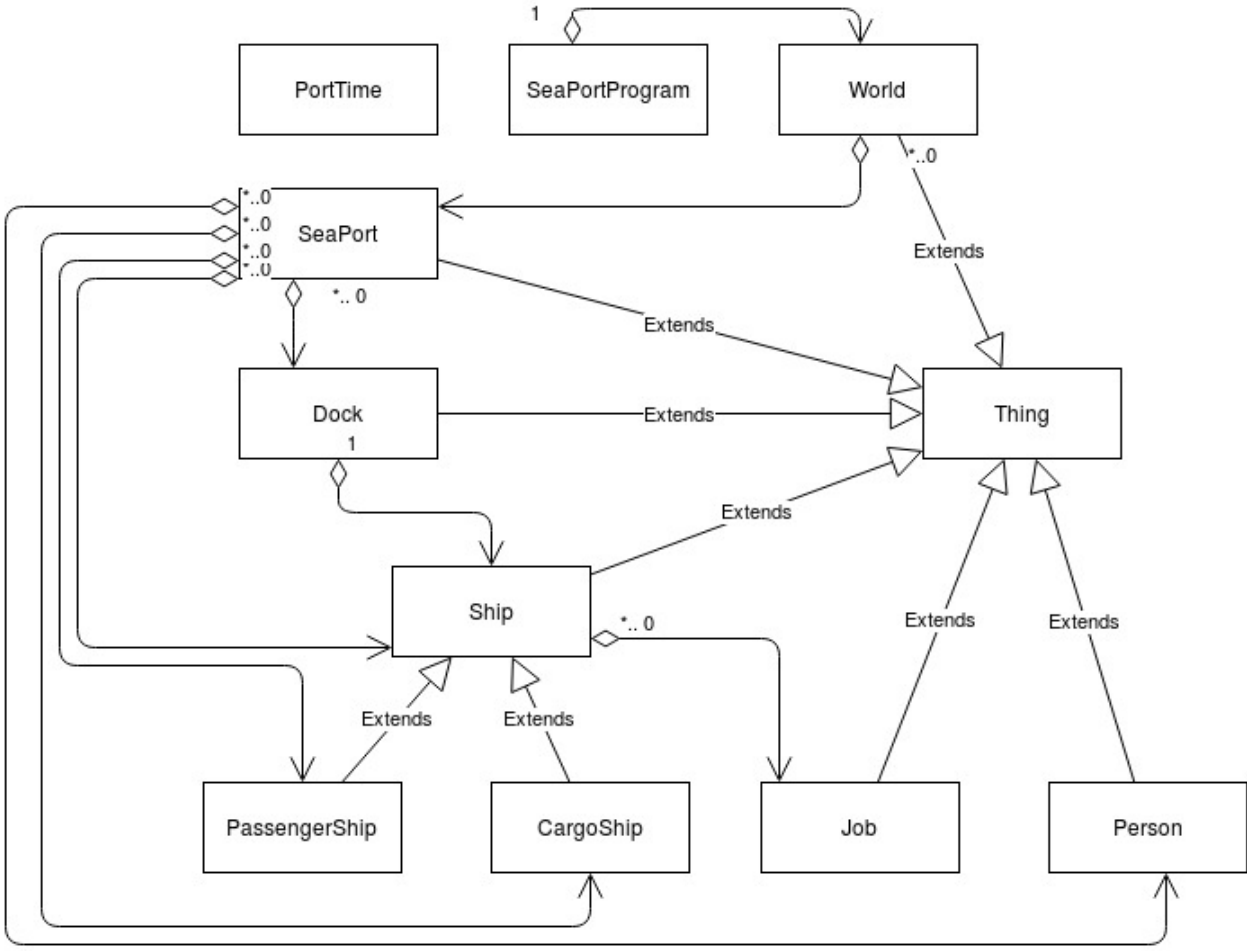
William Kendall

CMSC 335

Project 4

Resource Pools

UML


1
PortTime
SeaPortProgram
World
*..0
*..0
*..0
*..0
*..0
SeaPort
*.. 0
Extends
Extends
Dock
1
Extends
Thing
Extends
Ship
*.. 0
Extends
Extends
Extends
Extends
PassengerShip
CargoShip
Job
Person

Project 4 did not require any modifications to the structure of the program.

To elaborate on changes made, the Job class thread now is synchronized on people at the seaport. This was changed from synchronized on the seaport itself. Fixing the issue from project 3, the Job thread now starts when the Job object is created.

Gui changes since project 4 are cleaning up the GUI so it re-sizes nicely. The second change is an addition of a Jtable to show the workers in there current location, and the current job they are working on.

User Guide:

1) Opening a file

a) from the left side of the application, click the button “Open File”

b) from the open file dialog, select the data file and click the open button.

note: Jobs will start when the file is loaded. If a ship is in a dock, and the workers available for the job are present at the port, then the job will start

2) Pausing a job

At the bottom of the screen, the job progress table can be viewed; click the check box in the pause column in the row of the job that you wish to pause.

3) Canceling a job

At the bottom of the screen, the job progress table can be viewed; click the check box in the cancel column in the row of the job that you wish to cancel.

Other actions

Jobs that are currently running update the information panel, it is recommended to pause jobs when performing other actions.

4) Searching the database

Enter the name, index, or skill(of a person) in the labeled text box on the left hand side to search for the desired item. Click the ShowChildern button to view and child objects of the result. A dialog will be displayed with the result, or a message if no results can be found.

5) Sorting the queue

To sort the queue, click on a radio button by the type you would like to sort the queue by, then click the sort button.

Note: ships leave the queue and are moved into the dock, then removed from the dock when jobs are finished. This will result in the queue being empty when all ships have been moved to the docks.

Operation notes:

When a job is canceled or can not be completed due to the lack of workers with necessary skills, the job will be removed from the Job Progress table.

Test Plan:

Case	Test	Expected Results	Results	Comments
Open file	Selected test data file	Information from file displayed correctly and jobs start on worker availability	Pass	Assuming the data file is in correct format
No skill available	A job requires a diver, however there is no worker with this skill	Message displayed about no worker with skill, and job removed from jobs table	Pass	None
Searching	Search for name sara in the database	Message displayed with the person	Pass	None
Searching	Search for index 30000	Message displayed with the ship Gallinules	Pass	None
Searching	Search for skill diver	Message with no result found displayed	Pass	The diver skill is needed by a job, however there is not diver available.
Pausing	Pause job before running	Job does not start, and ship enters,	Pass	None

		but does not leave the dock.		
Pausing	Pause job while running	Job pauses, ship does not leave dock	Pass	Paused job will keep workers until the job is finished
Canceling	Cancel job before running	Message displayed, yes clicked, and job is removed	Pass	None
Canceling	Cancel job while running	Message displayed, yes clicked, and job is removed, workers released	Pass	None
Jobs start correctly	Observation	Jobs only start when the ship is in dock, and the the worker is in port	Pass	None

Lessons Learned:

Project 4 for me was about time management. I had a few ideas on how I wanted to get things done, however had to make some cuts in places so the over all project could get finished in the time I have set. Functionality wise, I felt like project 4 is an extension of project 3 with significant changes to the thread run function, however it does work in the way that project 3 did.

Lessons Learned From all Projects:

I think the Object oriented design of the project really shines when returning to the project over and over again with more changes and implementations. I have realized this by project 2 and tried to write code that could be modified and changed quickly. By this, that parts of the code are easy to track down and modify or extend. I found myself struggling many times with code that was difficult to make changes to and needed to rewrite them many times, but spending a little time making it more universal saved me time, allowed me to reuse my own code, and find parts more readily.

I really struggled with the gui and how the classes and gui should interact with each other. I am still a little unsure if I am doing things correctly, but feel that I have a good understanding of the reason and why I am doing things.