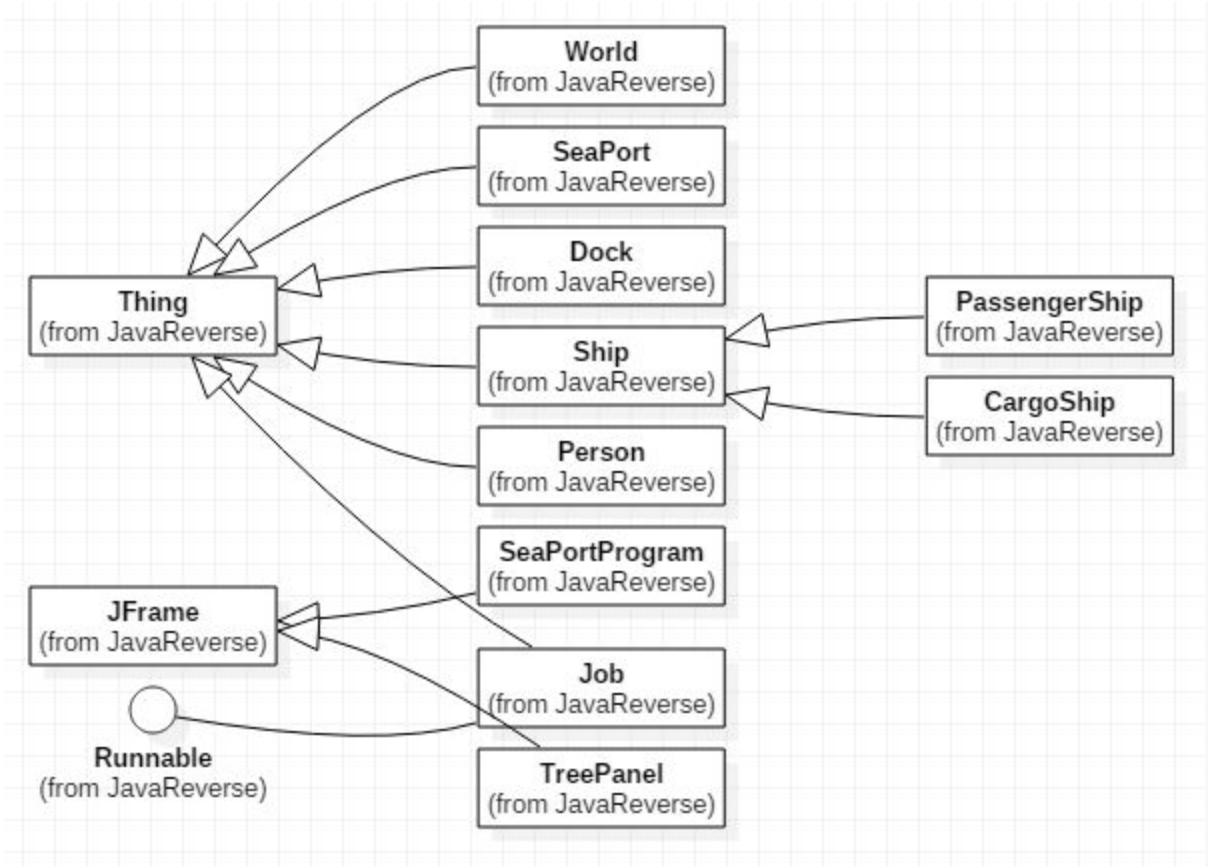
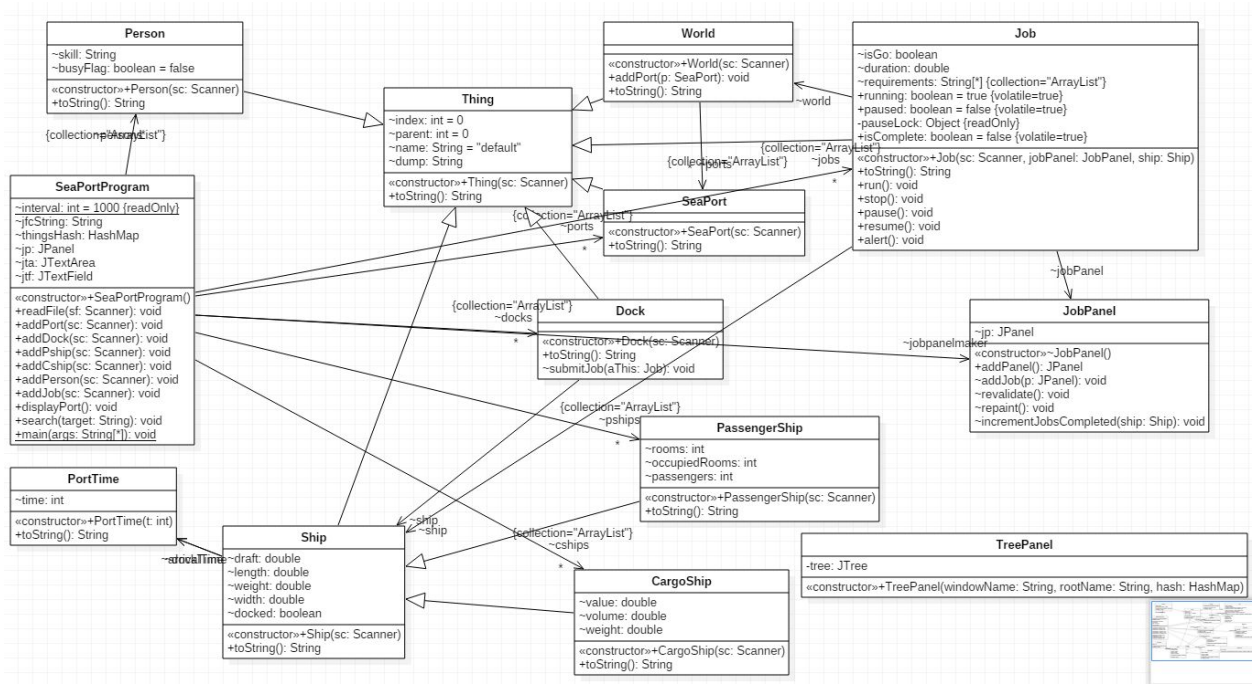
Design

Simple UML Flow diagram



More Detailed UML Diagram



Both of these PNG files are included in zip file in case they are hard to read

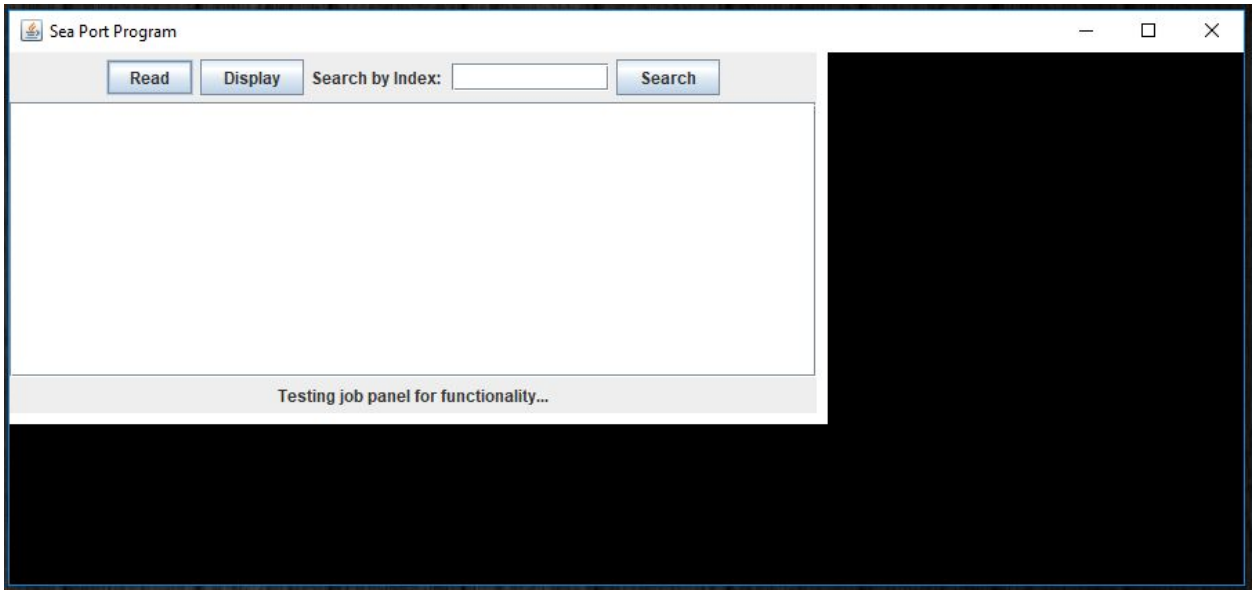
User's Guide

1. Open the JAR file named “Project” (this opens the program)
2. Click the “Read” button (this creates a separate GUI with the data in a sorted JTree) as well as creates the jobs needing to be done (only viewable by adjusting the size of the GUI after hitting button
3. Click the ‘key’ to drop down the needed object to view index and parent within the separate JTree GUI that was created
4. Click ‘Display’ to view all of the SeaPort information
5. Once you have located the index of the object you want to know more about in the JTree, insert that index in the text field labelled ‘Search by Index’ then click ‘Search’ (this will clear the text area and fill in the information you want.
6. You can view the jobs that should be done here, but the buttons associated with each job are still not functioning correctly

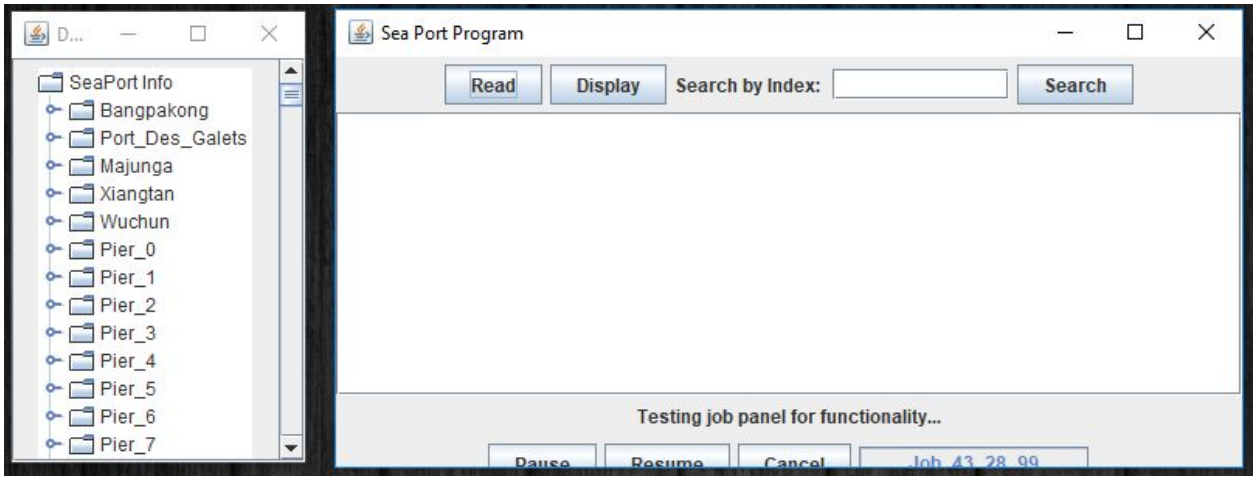
Test Plan

For testing I adjusted the code multiple times to test for different kinds of functionality due to errors occurring within the Job Class. The best I could get it to is what it is now in its current form with the section of code in the Job Class that handles the threading noted out. When I have this part as not noted out, my program shows the part of the GUI outside of the regular objects to be totally filled in with black, which I will do one test plan to show what I mean. As far as the rest of it goes, since I was unable to get this part of the code working correctly I was unable to fulfill some of the needed requirements concerning concurrency in this project. I believe I was close to getting there as many of the data lists and such are being filled in correctly, which I did check by looking at the different variables while debugging.

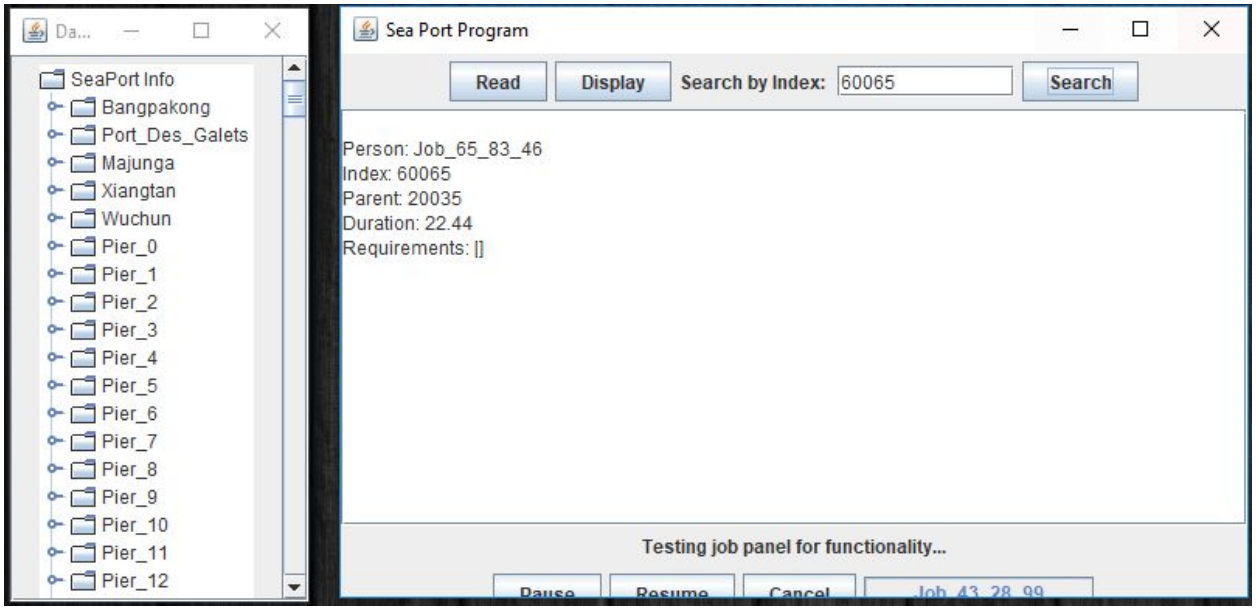
Test 1 - Showing that the program compiles correctly but when attempting to run, gets stuck in an endless loop somewhere and just fills the GUI with black as if it was filled in msPaint everywhere outside of the regular GUI objects.



Test 2 - After noting out the section of code that was causing the endless loop in test 1, the program compiles successfully, but this time doesn't get stuck, but finishes its tasks. The Jobs that are added to the GUI are not as they should be, but at least they get added in some way showing that those resources are available to use. Both Test 1 and Test 2 used the 'Test2.txt' data file which had Jobs included in it.

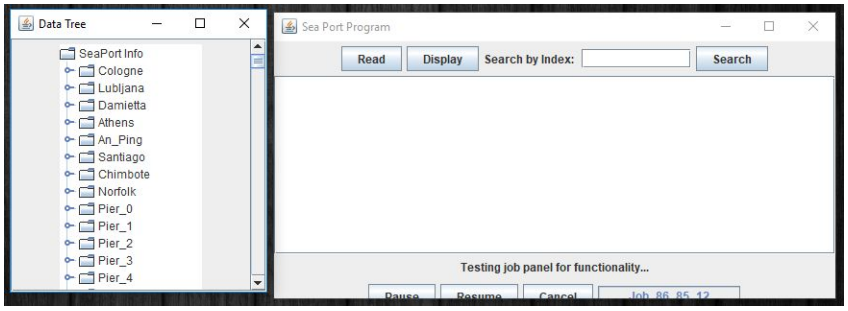


Test 3 - This test is simply showing that the Search function is still doing as it should which allows the user to insert an index to narrow down their search of the objects information.

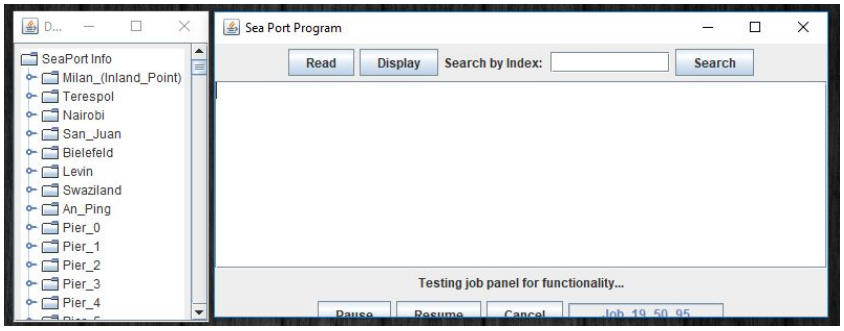


Test 4 - This test was done to show that the program continues to work even when using different data files. In previous tests we just used the file “Test2.txt”. In this test I will include three screen caps, each from a different file being used including: “Test3.txt”, “Test4.txt”, and “Test5.txt”.

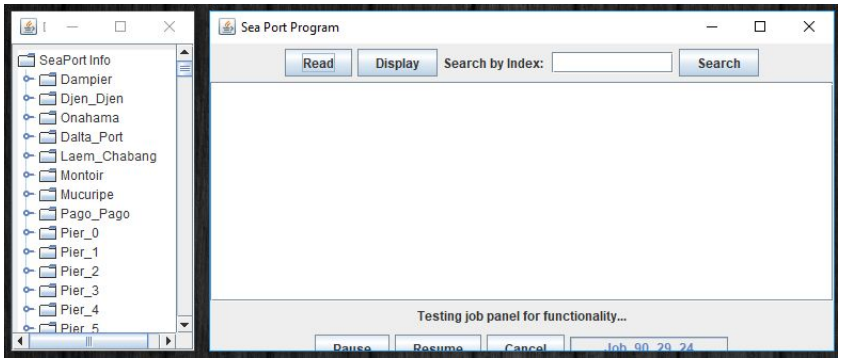
Test3.txt



Test4.txt



Test5.txt



Lessons Learned -

I learned a great deal during this project including more about how objects built within other objects function and also more about how threading works, even if I didn't get to a point to where the program was working as it should, at least it was showing at many points some of the correct information. Throughout the entirety of the class, constant having to go outside of the class materials to help me learn have made me feel like a much better programmer than I was when I began this class. In that way, this class and this project have been very beneficial to me. I learned as well that a very good tool to use when a program is not working as it should is to just hardcode the data that you need into the program to get it semi working, then work backwards from there to try and get the necessary information to go where it needs to to avoid having to hardcode it in. This helped a lot when I was debugging as I needed the program to at least compile before I could really debug effectively and check variable values. I will for sure be using this method in the future.