Daniel Zeltser

Student ID: 216446742

Course: EECS 3311 Section A

Project: Jframe Shapes

TA: eecs3311tafall2021@gmail.com

LAB REPORT

Introduction

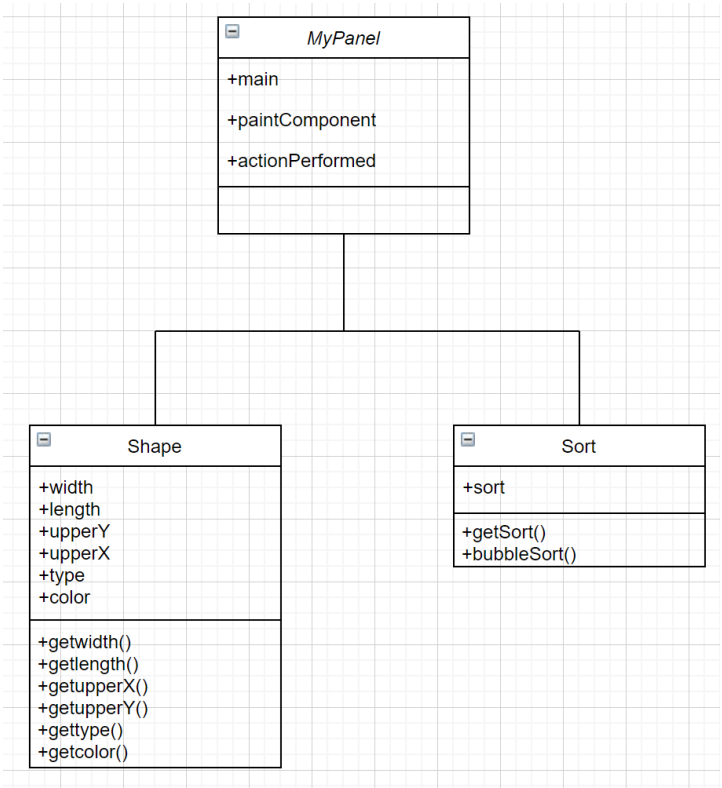
The project is all about using jframe to generate random shapes with random colours on an interface and then using the program to recognize the shapes so they are ordered by size from big to small.

The biggest challenge for me was at the beginning. This is the first time I was doing coding since last semester, and I forgot some of the basics. After a little while of sitting and reading I got back into it and started making some progress.
Another struggle I had is with implementing the sorting system, I had to remind myself by looking up a couple of sorting systems online and then figuring out how to implement it with my code.

The OOD principles that are used in this project are all about working with shapes and placing them into a user interface. In addition to placing them we also must move as the program is running and sort them. Sorting is another part of OOD that is seen in the program.

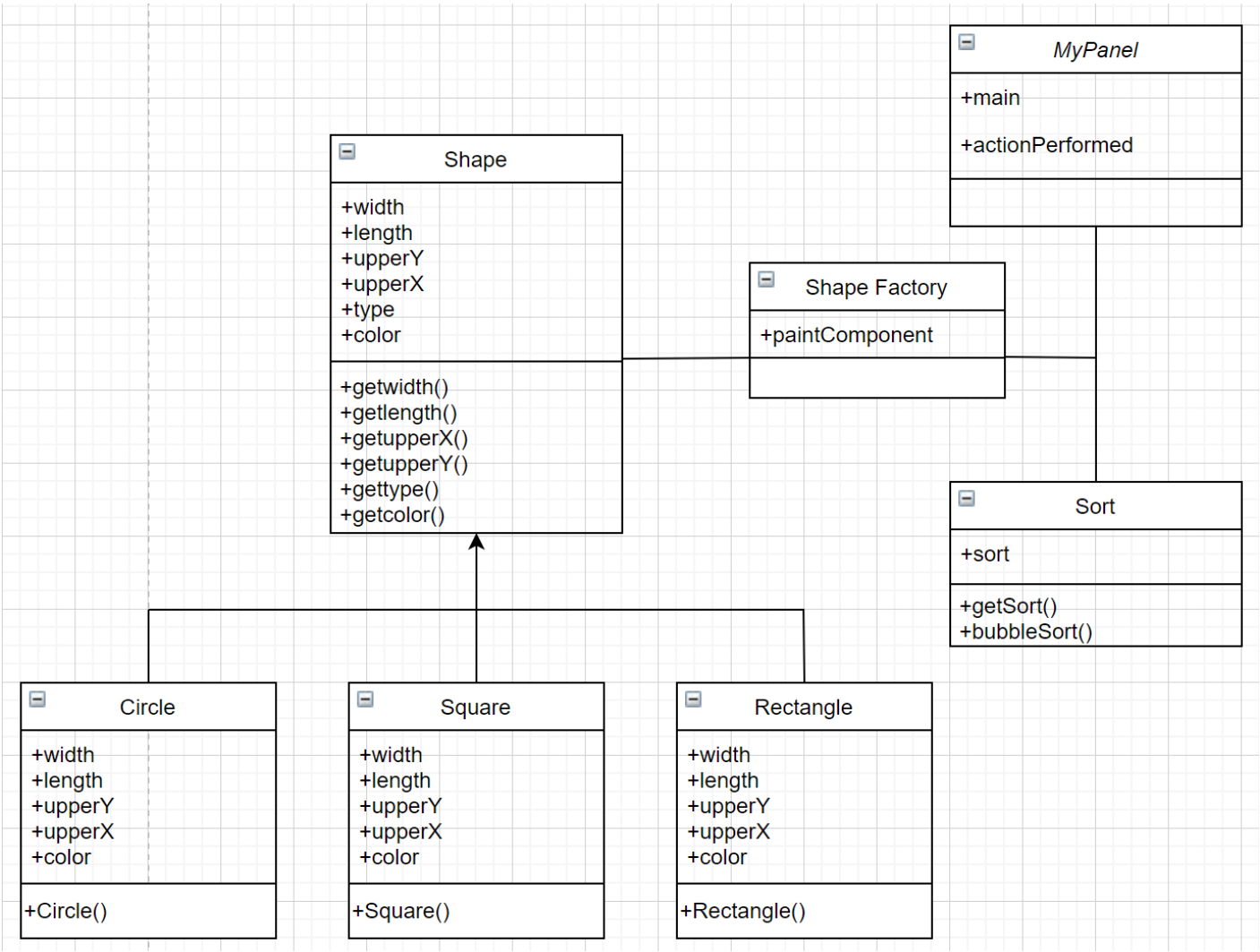
The report will be broken down into multiple parts. The introduction in which I will briefly go over the idea of the project, then the design part which is going to include a couple of diagrams explaining the thought process behind how the program is operating. The implementation part, in which I will discuss about how I made the various parts of the program work. And finally the conclusions I came up to after completing the program.

Design



That's the design pattern I ended up using. I have used the shape class to be differentiate between the shapes by using a type of attribute that allows to distinguish between each shape. This allows me to have less classes while still managing to have an easy way of knowing which type of shape I am working with.

After the random shape is chosen the shape gets a random color using the color paintComponent. Afterwards the is displayed on the UI using paintComponent aswell. And then finally the Sort class is used to go through the areas of all of the random shapes and sorting them by size using the bubbleSort method, which then send all of the shapes back to paintComponent and displays all of the shapes on the UI again, this time in decreasing order using their area which was calculated when the shape was created.



This is an example for an alternative design I could have went with. This design the benefit of having an easier way of adding shapes to the system just by adding another class. But the downside would have been that this was a bit more complicated to make intially in my opinion.

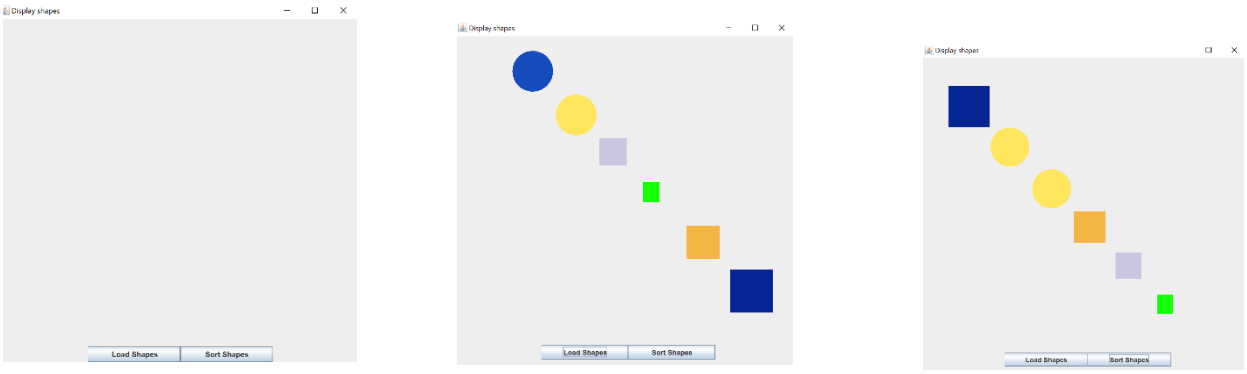
Implementation

For the sorting, I used a simple bubble sort algorithem, the algorithem is using all of the shape's areas as an array and then it goes throught the array and reorginzes it from big to small.

I ended up using the first class diagram, when I was implemeting it I stated off with the main and the paintComponent until I figured out the basics, then I moved into the shape class and kept adjusting the paintComponent method. Finally I moved into making the Sort class and further

adjusting the paint compenent and main method until I ended up with the working prototype. At the end I was going over the whole thing and making small adjustments.

The program was run on eclipse version: 2019-06 (4.12.0), jdk-12.0.2.



Here Is the UI of the program, at first its just an empty UI with two buttons, after clicking on Load Shapes the shapes appear. And after clicking on Sort shapes all of the shapes get ordered by area size.

Conclusion

The main thing that went well with the program was the sorting class. At first I was a bit worried that its gonna be really hard for me, but I ended up figuing it out a lot faster than I expected.

The main thing that went wrong in the project was at the beginning where I kept getting myslef worried over the project and I got a bit overwhealemd of thinking of all of the different things I had to do.

I have learn how to use jframe and that I actually enjoyed it quite a bit.

recommendations to ease the completion of the software project

1. Taking the project step by step and breaking it down to smaller parts.
2. Planning out an idea of how toa pproach the project.
3. To code more often to avoid forgetting the basics.