Name: Ishraque Mashrur

Student ID: 216930729

Course: EECS 3311

Section: B

Project: LAB 1

TA: Naeji Alireza

Part I: Introduction

The project is an application that displays 2 buttons. The first of which loads 6 randomized shapes on screen, and the second sorts these randomized shapes in increasing order of surface area and displays them on screen. The challenges associated with this project involve learning Java Swing and applying Object Oriented Design principles in order to achieve the results. In my project, I will use the 4 main principles of OOP, Inheritance, encapsulation, polymorphism and abstraction, along with the factory design pattern to structure my project.

Part 2: Design of the solution

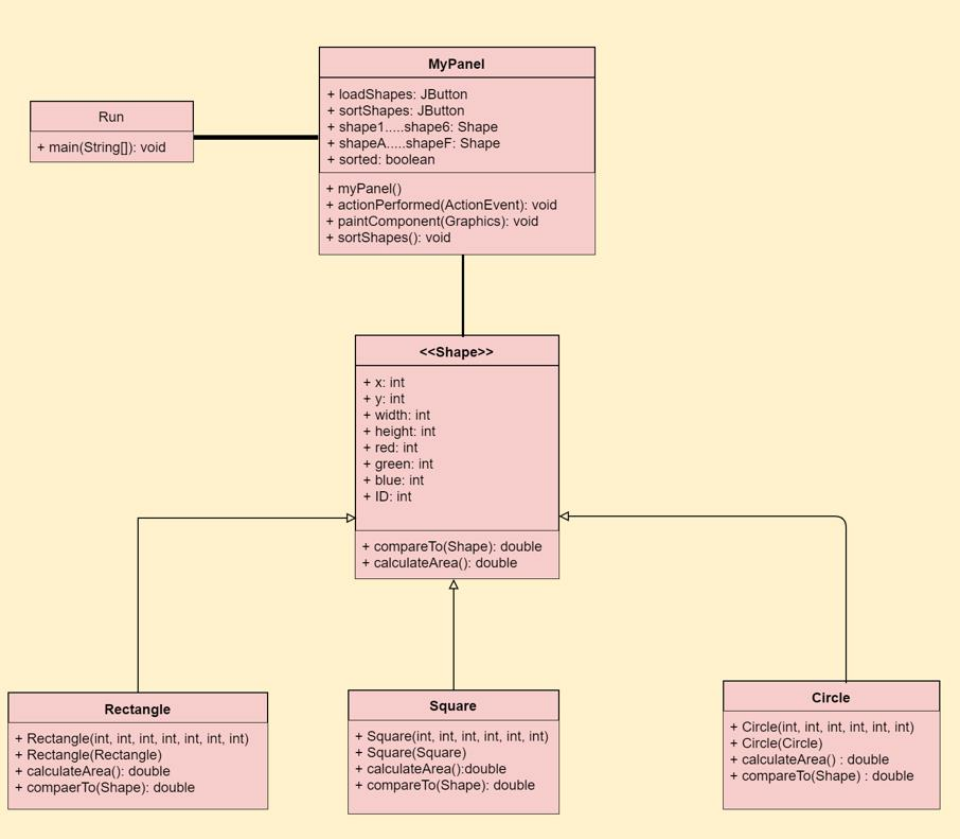


Fig: UML Diagram

In my project I've used encapsulation, abstraction by making the Shape class abstract, inheritance and polymorphism via the Shape class and its child classes.

Part III: Implementation of the solution

For my solution, I've used the insertion sort algorithm and stored each of the sorted shapes in order in 6 shape variables which I then displayed. Using the aforementioned OOP principles and factory design pattern I've compiled all of my classes in my class diagram in java, using the Eclipse IDE ver. 4.16.0 on JKD 15

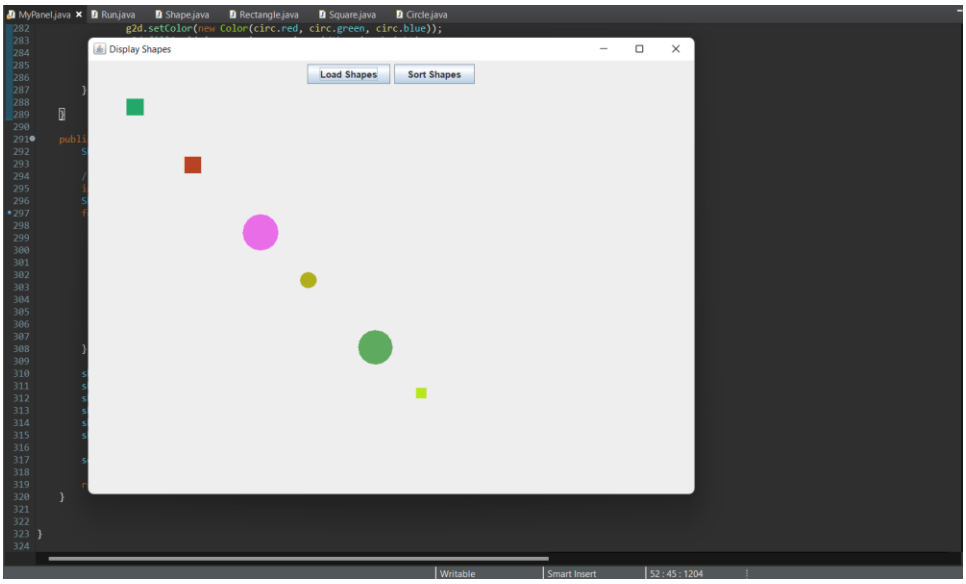
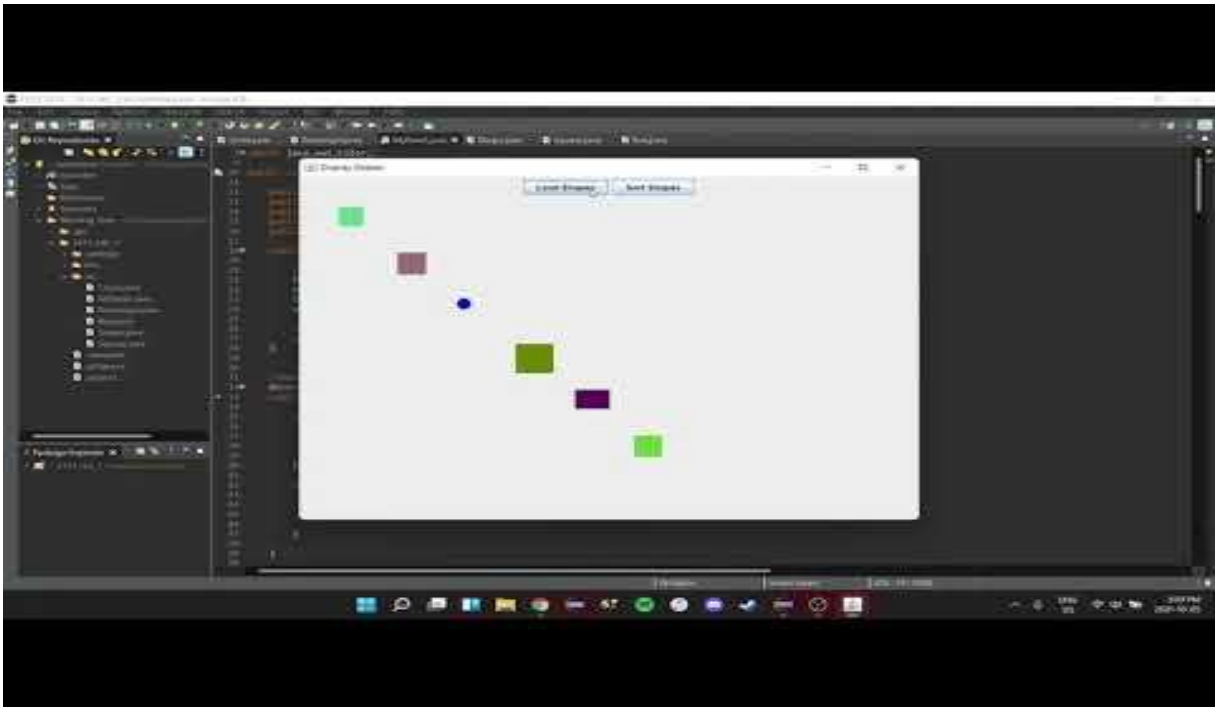


Fig: Snapshot of code execution

Link to video showing application running: [EECS 3311 Project 1 (Fall 2021)]



Part IV: Conclusion

What went well in the software project: The implementation of the UML in my project went fairly smoothly

What went wrong in the software project: Figuring out Swing and how to display everything correctly took the most significant portion of the time allotted for this project as many things had gone wrong forcing me to use clever workarounds instead

What I have learned in the project: How to apply OOD principles and patterns, how to create and implement UML diagrams, how to create User Interfaces in Java using Swing.

What are my top three recommendations to ease the completion of the software project?:

1) Make sure to have a clear idea of exactly what you want to implement and draw a UML class diagram before starting to write any code

2) Focus on getting the user interface done first before the other classes

3) Make sure there is equal distance between each shape and they are sized appropriately enough to fit on screen