EECS3311: Software Design

Lab 01

Omar Saeed

215911555

Report: Software project 1.

1. Part 1:

* Introduction:

This software project aims to create a simple application that can display 6 shapes and sort those 6 shapes according to their surface area. Screen displays 2 buttons: Load Shapes and Sort shapes, when the program is first executed. After engaging with load shapes button, 6 new shape objects appear on the screen, all having different sizes. When sort shapes is clicked, all 6 shapes on screen are sorted and redisplayed.

The biggest challenge in creating this application was getting the actionListeners working. Buttons would not respond to what they are supposed to do, when clicked. Another great problem was that the shape objects would appear on the Jframe as soon as the application was run.

In implementing this software, I have used various Object-Oriented principles. Starting off with the very core of the code: JPanel/Jframe. The main class of the function, containing main method(psvm), extends JPanel, which means that class main is inheriting all methods from JPanel, its parent, and is its subclass. Main class object is-a child of JPanel therefore it is-a JPanel itself. Using this, I added button and shapes to the JPanel and further added them to the frame.

This report will consist of 3 parts: Introduction, Design of the solution and Implementation of the solution. Introduction, this part, contains an overview of what will be featured in this report and some basic functionality of the program. Design of solution covers how different objects are working together, how different classes are interacting, etc. Implementation of the solution will have in-depth analysis of the code and its algorithms.

1. Part 2:

* Class Diagram:

Diagram

Description automatically generated

Main class uses 3 additional classes to create shape objects and displays them on the shape. There are 3 methods inside main class that create an object of type MyRectangle, MyCircle or MySquare. Inside these method constructors of each class is called and an object is created, which is stored inside an ArrayList. This ArrayList is later used by the method paintComponent that loops through these ArrayLists and displays every object stored inside it.

1. Part 3:

* Sorting Algorithm:

The sorting algorithm implemented works by swapping adjacent elements if they’re not in required order. This algorithm starts with a for loop that runs the length of objects present in the array. It compares the surface area of the current object with the next object. If the current object’s surface area is smaller than next object, then our algorithm swaps these two objects.

The code has been exported into an executable java file and runs as a java application. Main class is a child of JPanel hence JPanel has been imported into the package. All 3 shape classes implement comparable interface.

This program has been developed using Eclipse: Version: 2020-06 (4.16.0). And Java: java version "14.0.1" 2020-04-14

* Screenshot of application:

Chart, waterfall chart

Description automatically generated

1. Conclusion:

The explanation of what was required from the project was very clear, and it helped me implement the code easily. This part went the best in this assignment.

My time management was what went wrong in this project. I underestimated how much time was required to code this and write the report which led to the whole project being rushed towards the end.

After this project I have learned how to create class diagrams and write detailed reports on code.

Top 3 recommendations to ease it:

1. Better time management
2. Better understanding of JPanel/JFrame
3. More familiarity with Github.