Meetkumar Patel – 216814352

EECS 3311 Section B

Naeiji Alireza

Shape Sorting Project

Part 1: Introduction

The Shape Sorting Project is a to create a GUI based application. The main motive of this project is to display six shapes, of three different types on the screen. Loading is done when user makes a request via a button. Three different types of shaped are square, rectangle and circle. The size of the shapes is not defined. However, the shapes are not allowed to overlap each other. The colors of the shapes are also undefined. On every button click, a new combination of color and size is loaded. The other function of this application is to Sort these shapes based on their Area when user makes a request via a button.

The main objective for creating this project is to understand some of the most important concepts like Object Oriented Design principles, design pattern, using Java language effectively. This project also requires knowledge of Sorting Algorithms.

Object-oriented programming combines a group of data attributes with functions or methods into a unit called an "object." Object Oriented Programming has four basic concepts:

1. Encapsulation: The mechanism of wrapping the data and the code acting on the data together.
2. Abstraction: The process hiding unnecessary details and exposing only essential information to user.
3. Inheritance: When an object acquires the properties of the parent object, it is known as inheritance.
4. Polymorphism: The ability of any data to be processed in more than one form.

Object oriented programs are easy to reuse and offer a clear modular structure for programs.

Design Patterns usually help assigning responsibilities to objects. The pattern used to design this project are Factory and Singleton. Both these patterns are creational and they focus on the instantiation of objects.

Factory Pattern: This pattern offers a way to instantiate objects of subclass whose parent class has several subclasses. It allows creating objects dynamically using a common interface. Singleton Pattern: This pattern ensures that a class has only one instance and provides a global point of access to that instance.

This report includes UML diagrams with explanation of implementation, how OOPs are used, Sorting algorithm, and conclusion, in a sequential manner. I It also includes screenshots and a short video of the working of the application is also submitted, highlighting all functionality.

Part 2 : Design

Diagram, schematic

Description automatically generated

**Part 3: Implementation of the solution**

*Sorting technique:*

The program requires the use of a sorting algorithm in order to arrange the displayed shapes in ascending order. There any possible ways to do that. The one that is been used is Bubble Sort.

Bubble sort is a very simple comparison-based algorithm. Each pair of adjacent elements is compared. Based on their order, they are swapped. As compared to other algorithms this algorithm is slow. It is considered not suitable for large data set. It has average and worst case complexity are of O(n2) where n is the number of items. Since the number of elements are just six, this algorithm works absolutely fine in this program. Insertion sort or selection have the same time complexity. However, both of them are unstable. In order to achieve stability divide and conquer algorithms like merge sort can be used. Megre sort has time complexity of O (n log n) in both worst and average case.

*Compilation and implementation:*

In order to implement this project, Eclipse 2020-2019 IDE with Java Se Development Kit 15.0.02. was used however, this program can run on device which has a Java Development Kit.

*Snapshots of Execution:*

Given below are the screenshots of the application.

Starting of the application:

Graphical user interface, application, Word

Description automatically generated

On Load Shapes Button Click:

Chart, waterfall chart

Description automatically generated

On Sort Shapes Button Click:

Chart, bubble chart

Description automatically generated

Part 4: Conclusion

This project exposed me to many new things. I had never created any Gui application. This was the first working with JFrame. I am so happy that this project compiled successfully. Also understanding the Sorting algorithm was very challenging for me. I tried to implent the merge sort algorithm at first since it has a better time complexity. However, I was unable to implement it. This was the part that went wrong for me. So I implemented bubble sort.

Through this project I was able to learn creational design patterns very deeply. I learned that there are many was to create a singleton class, which is very cool. The concept of time complexity was also new to me.

My top three recommendation would be:

1. Practice making UML diagrams
2. Try to implement those diagrams and then analyze it
3. Learn other patterns as they are very useful