## FINAL PROJECT REPORT

**PROJECT NAME: CALCULATOR**

Course code: CSE214/215

### Course title: Object Oriented Programming with Lab

Submitted to,

##### Teacher Name: Hasnur Jahan

##### Designation : Lecturer

Daffodil International University

Submitted by,

Student Name: Tamanna Akter

ID: 221-15-5298

Section: 61\_M

Department of CSE

**Date: 07 June 2023**

**INDEX:**

The Project File contains the following contents-

1. Introduction………………………………………………………3
2. Objectives………………….…………………………………….3
3. Technologies Used……………………………………………….3
4. Project Implementation………………………………………….3
5. Project Overview/Summary……………………………………..4

9. Limitations of the Project………………………………………. 5

10. Future Scope……………………………………………………6

11. Conclusion………………………………………………………7

12. Screenshot………………………………………………………8

**About:**

Aim of the Project Nowadays the calculators that we use are so advanced. They range from scientific to graphing. Technology has come such a long way. Now everything is available on the computer, so why not calculators right? Due to the Java programming language, we are now able to create all sorts of applications. We have created a simple addition calculator. It calculates the total of each of the integers which you input, all added together. Calculators are one of the most useful tools in life, although our simple addition calculator will not be used to calculate taxes, it will indeed serve the purpose of adding a few numbers. When you actually run the program, it looks pretty simple right? Well itis not as simple as it looks. There is a lot going on “behind the scenes” of the calculator, which is where all the coding lies. What makes the calculator look the way it does? What makes the calculator work? All of the questions lie behind the coding of our program.

**Introduction:**

The Calculator GUI project is a Java-based application that provides a graphical user interface for performing basic arithmetic calculations. The project aims to create an intuitive and user-friendly calculator that can handle addition, subtraction, multiplication, and division operations.

**Objectives:**

The primary objectives of the project are as follows:

Develop a calculator application with a graphical user interface.

Implement basic arithmetic operations such as addition, subtraction, multiplication, and division.

Provide a responsive and intuitive user experience.

Ensure error handling for invalid inputs and division by zero scenarios.

Utilize Java's Swing library for creating the graphical user interface.

**Technologies Used:**

The project utilizes the following technologies and tools:

Java Programming Language: Java 8 or higher.

Swing Library: To create the graphical user interface components.

Integrated Development Environment (IDE): Any Java-supported IDE such as Eclipse or IntelliJ IDEA.

Version Control: Git (optional but recommended).

**Project Implementation:**

The calculator GUI application is implemented in several steps:

1.Setting up the Project:

Create a new Java project in your preferred IDE.

Set up a new class file to contain the main application code.

Import the necessary Java Swing packages for creating GUI components.

2. Designing the User Interface:

Create a JFrame (window) object to hold the calculator interface.

Set the title, size, and layout of the JFrame.

Add text fields, buttons, and labels to represent the calculator's display and input elements.

3. Implementing the Calculator Logic:

Define variables and fields to hold the calculator's current state and input values.

Implement action listeners for the number buttons (0-9) and arithmetic operation buttons (+, -, \*, /).

Handle button clicks to update the display and perform calculations based on user input.

Incorporate error handling for invalid inputs and division by zero scenarios.

4. Testing and Debugging:

Test the calculator application by performing various calculations and verifying the results.

Debug any issues or errors encountered during testing.

Refactor the code as necessary to improve readability, maintainability, and performance.

**Challenges and Lessons Learned:**

During the development of this project, several challenges and lessons were encountered, including:

Designing an intuitive and visually appealing user interface.

Handling user input and ensuring the correct order of operations for

calculations. Implementing error handling to prevent program crashes or incorrect results. Enhancing code modularity and reusability by applying object-oriented principles.

**Project Overview/Summery:**

The project is a JAVA application which combines the knowledge of programming and mathematical science into one. It makes use of a famous Java IDE (Integrated Development Environment) - NetBeans IDE 8.0.2 which is currently under oracle's development.

NetBeans includes drag and drop GUI creation, thus making it much easier to develop the program. The project came into its existence by the following steps-

• Designing the algorithm of the application. It required application of scientific as well as mathematical concepts.

• Designing the layout/view/GUI (Graphical User Interface) of the application. This part provides the "LOOK" of the application that is visible to the user end.

• Coding the applications event-handlers (The code that should run in case the user presses a button the application's screen.)

• Debugging the application by testing it through various mathematical examples.

• Documentation which includes the project report and presentation.

.

**Limitations of the Project:**

The four developers of the application were unable to resolve the following

limitations/bugs-

* Does not calculate complex roots of nth degree polynomial
* Few libraries used are outdated and does not support new GUIS.
* Presently it can only be used by Engineering students only.

.

**Future Scope**

* App will be modified with Commerce Formulae, so that it can reach out more people.
* It will be available on Google Play Store and app store, for the students after complete development.
* People will be able to make graphs using this programmed.
* It will always be free to use.

**Conclusion:**

The Calculator GUI project successfully created a Java-based calculator application with a graphical user interface. It enables users to perform basic arithmetic operations in a user-friendly manner. The project achieved its objectives by utilizing Java's Swing library and implementing the necessary logic for handling user input and calculations.

Future enhancements to this project may include adding support for advanced operations such as square root, exponentiation, and memory functions. Additionally, integrating unit testing frameworks and conducting comprehensive testing will help ensure the calculator's accuracy and reliability.

By completing this project, valuable experience and knowledge were gained in GUI development, event handling, error handling, and Java programming in general. This project serves as a solid foundation for future GUI-based application development in Java.

**References:**

Java Swing Documentation: https://docs.oracle.com/en/java/javase/14/docs/api/index.html

Note: The content above provides a general outline for a project report on a calculator GUI application in Java. Feel free to add or modify sections based on your project's specific requirements and finding.

**Screenshot:**



