# Final Report:

## 1. Introduction:

This final report presents the completed implementation of the Calculator Web Application.
It evaluates how well the implementation matches the analysis and design objectives, and summarizes achieved results.

## 2. Implementation Summary:

### The calculator was implemented using:

- HTML for structure.
- CSS for layout and styling.
- JavaScript for logic and functionality.

### The system is fully functional and supports real-time calculations.

## 3. Requirement Satisfaction:

### A: Functional Requirements:

- Numeric entry.
- Operator entry.
- Expression calculation.
- Result display.
- Clear button.

### B: Non-Functional Requirements:

- Fast performance.
- High usability.
- Reliable error handling.
- Maintainable code.

# 4. System Features:

- Supports addition, subtraction, multiplication, division.
- Provides real-time calculation.
- Supports decimal operations.
- Prevents application crashes.
- Uses a nice UI design.

# 5. Testing:

I tested the project by using the common QA ways of testing. Invalid expressions, edge cases and empty calculations

# 6. Challenges & Solutions:

- Challenge: Handling invalid expressions
  Solution: Implemented error handling using try–catch

# 7. Conclusion:

- The Calculator System meets course expectations and fulfills both functional and non-functional requirements.
- It demonstrates a complete software engineering workflow including analysis, design, and implementation.
- It's simple and didn't much work on the project itself but rather focused in applying the concepts and what we learned in the course that the professor explained.
- Hence the simplicity allowed us to understand the way that reports are made and we understood the blueprint of the project, since that was the goal of the professor.

## 8.Future enhancements may include:

- History feature
- Backspace button
- Percentage support
- Keyboard support
- Including more complicated calculations such as Sin and Cos.