

Functional Document

1. Project Overview

This project implements a simple command-line calculator application in Java. The application allows users to perform basic arithmetic operations such as addition, subtraction, multiplication, and division on two numbers. The calculator is designed for straightforward use and provides immediate results to the user.

2. Features

1. Addition: Users can add two numbers.
2. Subtraction: Users can subtract one number from another.
3. Multiplication: Users can multiply two numbers.
4. Division: Users can divide one number by another.
5. Error Handling: The application includes basic error handling for division by zero and invalid operator input.

3. User Workflows

1. Start the application: The user executes the Calculator.java program.
2. Enter the first number: The application prompts the user to enter the first numerical value.
3. Enter the operator: The application prompts the user to enter a valid arithmetic operator (+, -, *, /).
4. Enter the second number: The application prompts the user to enter the second numerical value.
5. View the result: The application calculates and displays the result of the operation.
6. Application termination: The application closes after displaying the result.

4. Inputs/Outputs

1. Inputs:

1. First number: A double-precision floating-point number.
2. Operator: A character representing an arithmetic operation (+, -, *, /).
3. Second number: A double-precision floating-point number.

2. Outputs:

1. Result: A double-precision floating-point number representing the outcome of the arithmetic operation.
2. Error messages: Textual messages indicating issues such as "Error: Division by zero error" or "Operator is invalid".

5. Business Rules

1. Valid Operators: Only +, -, *, / are recognized as valid arithmetic operators.

2. Division by Zero: Division by zero is not permitted and will result in an error message.
3. Numerical Input: All number inputs must be valid double-precision floating-point numbers.
4. Single Operation: The calculator performs one operation per execution.

6. Error Handling

1. Division by Zero: If the user attempts to divide by zero, the application will display "Error: Division by zero error" and terminate.
2. Invalid Operator: If the user enters an operator other than +, -, *, or /, the application will display "Operator is invalid" and terminate.
3. Invalid Number Input: The current implementation relies on Java's `Scanner.nextDouble()` for number parsing. Non-numeric input will cause an `InputMismatchException`, which is not explicitly handled in this version.