
<Company Name>

<Kool Kalcuator>
Software Requirements Specifications

Version <1.0>

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

Revision History

Date	Version	Description	Author
<08/28/23>	<0.5>	<Started work on the completion of this document>	<David, Ky, Ian, Noah, Humzeh>
<10/05/23>	<1.0>	<discussed and finished out the document>	<David, Ky, Ian, Noah, Humzeh>

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

Table of Contents

1.	Introduction	4
1.1	Purpose	4
1.2	Scope	4
1.3	Definitions, Acronyms, and Abbreviations	4
1.4	References	5
1.5	Overview	5
2.	Overall Description	5
2.1	Product perspective	5
2.1.1	System Interfaces	5
2.1.2	User Interfaces	5
2.1.3	Hardware Interfaces	5
2.1.4	Software Interfaces	5
2.1.5	Communication Interfaces	Error! Bookmark not defined.
2.1.6	Memory Constraints	5
2.1.7	Operations	5
2.2	Product functions	6
2.3	User characteristics	6
2.4	Constraints	6
2.5	Assumptions and dependencies	6
2.6	Requirements subsets	6
3.	Specific Requirements	6
3.1	Functionality	6
3.1.1	<Functional Requirement One>	Error! Bookmark not defined.
3.2	Use-Case Specifications	7
3.3	Supplementary Requirements	7
4.	Classification of Functional Requirements	7
5.	Appendices	8

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

Software Requirements Specifications

1. Introduction

This software is made to be able to do basic calculations and identify any syntax errors. The main calculations we will be doing are adding, subtraction, multiplying, dividing, using powers, and parentheses. This document will go more in depth into what this software has to offer, including how each operation works and the limitations entailed. This document will contain descriptions of functionalities intended for this software project. Understanding the functionality of the project will allow the group to create an effective program.

1.1 Purpose

The Software Requirements Specification document outlines the necessary components of our project. This includes interfaces with the user and other systems, constraints, and a detailed description of all functionalities. This software will be able to run on regular computers for the purpose of showing what we have learned and achieved in class (EECS348).

1.2 Scope

The breadth and width of the project includes the functionality and potential audience of the project. We intend to create a program that can interpret and output solutions to arithmetic statements. We will be able to catch mathematical errors like dividing by 0, and syntax errors like forgetting a bracket or adding an extra sign and many more. This program is intended to function as a single instance of the program without networking outside of the user's device.

1.3 Definitions, Acronyms, and Abbreviations

Definitions:

- **Mathematic Operator:**
 - Symbol used for specifying relationship and transformation of numbers
- **Order of Operations:**
 - The specific order in which arithmetic statements are solved given their operation and position.
- **Arithmetic:**
 - Mathematical operations that perform work upon real numbers of the real number line. One of the oldest forms of math.
- **Addition (ADD):**
 - Operation that combines two numbers to find their sum.
- **Subtraction (SUB):**
 - Operation that finds the difference between two numbers.
- **Multiplication (MUL):**
 - Operation that calculates the product of two numbers.
- **Division (DIV):**
 - Operation that divides one number by another to find the quotient.
- **Exponentiation (POW):**
 - Operation that raises a base number to a specified power.
- **Parentheses (BRACKETS):**
 - Symbols "(" and ")" used to group expressions and control the order of operations.
- **Syntax Error:**
 - An error that occurs when the input expression is not formatted correctly or violates the

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

rules of the calculator's syntax.

- **Uncalculatable Error:**
 - An error that occurs when the calculator cannot perform a calculation, often due to dividing by zero or taking the square root of a negative number.

Acronyms:

- **PEMDAS:**
 - Parenthesis, Exponential, Multiplication, Division, Addition, Subtraction
- **GUI:**
 - Graphical User Interface

1.4 References

The project requirements provided by the instructor will dictate large amounts of the functional requirements for this project.

1.5 Overview

The following pages describe the use cases, constraints, and functionality of our basic calculator. Interfaces, operations, basic functionality, and various constraints will be explored in depth. Section 2 of our document gives an overall description of the project requirements. Section 3 goes more in depth in describing what each requirement is. Section 4 goes requirement by requirement and assigns them importance. Overall, this outlines the requirements for constructing this project.

2. Overall Description

2.1 Product perspective

This calculator software is a standalone application that allows users to perform arithmetic and mathematical operations on equations entered by the user. This software operates independently and doesn't require any external system or software.

2.1.1 System Interfaces

N/A

2.1.2 User Interfaces

- User inputs the equation via input line

2.1.3 Hardware Interfaces

N/A

2.1.4 Software Interfaces

- Interface with C++ Math library

2.1.5 Memory Constraints

- Allocated memory by average user PC

2.1.6 Operations

- Addition
- Subtraction
- Multiplication
- Division
- Exponentials

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

- Parenthesis handling
- Take equation from user as input
- Return solution or error

2.2 Product functions

- Evaluate Equations
- Handles Input Errors

2.3 User characteristics

- Capable of using a computer
- Understands basic arithmetic
- Literate

2.4 Constraints

- Must be coded in C++
- Must be completed by due date
- Must be uploaded to GitHub
- Must contain user manual

2.5 Assumptions and dependencies

- We assume that we will have enough time to complete the project
- We assume that requirements will not change
- We are dependent on the constancy of the laws of arithmetic
- We assume users have basic knowledge of arithmetic operations

2.6 Requirements subsets

- Expression Parsing
- Operator Support
- Handling Parenthesis
- Calculation Order
- Error Handling

3. Specific Requirements

3.1 Functionality

3.1.1 Expression Parsing

- Program must parse expressions with relative mathematical rules considered

3.1.2 Operator Support

- Program must be able to recognize and handle the following operations: Addition (+), Subtraction (-), Multiplication (*), Division (/), Modulo (%), and Exponentiation (^)

3.1.3 Handling Parenthesis

- Program must recognize precedence determined by parenthesis, and evaluate according

3.1.4 Calculation Order:

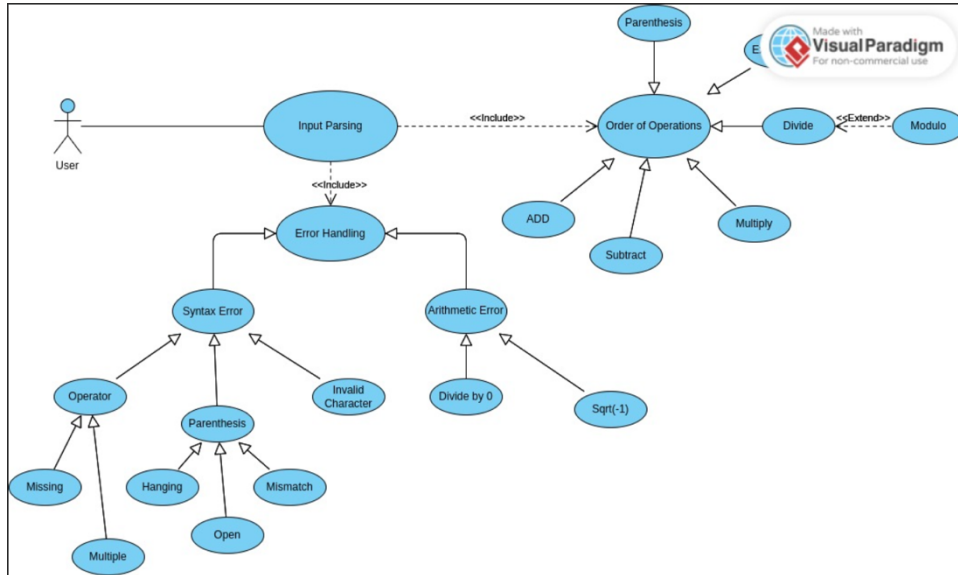
- Decides which operation should be calculated first while going through the equation. Brackets > exponential > multiplying, dividing, or modulo > and finally adding or subtracting
- Operations who have the same priority will be solved from left to right

3.1.5 Error Handling:

- While parsing checks for any form of error. Examples include too many operations signed, too little operation signs, dividing by zero, square root of a negative.
- Informs the user about the error and lets the user adjust the equation accordingly.

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

3.2 Use-Case Specifications



3.3 Supplementary Requirements

The project will be coded and compiled in C++. Any GUIs created for user interactions will also be coded in C++. The code will be maintained on a GitHub repository.

4. Classification of Functional Requirements

Functionality	Type
Addition	Essential
Subtraction	Essential
Multiplication	Essential
Division	Essential
Exponents	Essential
Parenthesis	Essential
Order of Operations	Essential
Arithmetic Error	Essential
Syntax Error	Essential
User input arithmetic statement	Essential
Output solution to arithmetic statement	Essential
Return invalid equation to user after error	Desirable
Fix errors	Optional
View equation history	Optional
Modulo	Essential
User-friendly GUI	Optional

<Project Name>	Version: <1.0>
Software Requirements Specifications	Date: <10/09/23>
<document identifier>	

5. Appendices

- Kool Kalkulator Management Plan