
Bonsai Tree

M.A.R.C.O.
Software Requirements Specifications

Version 1.1

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

Revision History

Date	Version	Description	Author
15/10/2023	1.0	Initial document creation.	Pashia Vang
01/12/2023	1.1	Revised flow chart	Marco Martinez
03/12/2023	1.1	Revised descriptions	Mick Torres

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

Table of Contents

1.	Introduction	4
1.1	Purpose	4
1.2	Scope	4
1.3	Definitions, Acronyms, and Abbreviations	4
1.4	References	4
1.5	Overview	4
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	5
2.1.6	Memory Constraints	5
2.1.7	Operations	5
2.2	Product functions	5
2.3	User characteristics	5
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	<Addition>	6
3.2	Use-Case Specifications	7
3.3	Supplementary Requirements	7
4.	Classification of Functional Requirements	7
5.	Appendices	8

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

Software Requirements Specifications

1. Introduction

The introduction section of this Software Requirements Specification (SRS) offers an overview of the document, providing insights into its purpose, scope, definitions, acronyms, abbreviations, references, and an overall perspective of the SRS.

1.1 Purpose

This Software Requirements Specification (SRS) describes the external behavior of the application M.A.R.C.O., as well as its nonfunctional requirements, constraints, and other information to provide a complete description of the requirements for the software.

1.2 Scope

This section outlines M.A.R.C.O.'s purpose. It's a calculator for fundamental math operations, including addition, subtraction, multiplication, division, exponentiation, and modulo. Users input math problems via a text-based interface. This section defines what M.A.R.C.O. can do.

1.3 Definitions, Acronyms, and Abbreviations

- SRS: Software Requirements Specification
- M.A.R.C.O.: Multi-Arithmetic Righteously Calculated Operations

1.4 References

N/A

1.5 Overview

The SRS contains an overall description of the application that includes the interfaces, requirements, and constraints, as well as descriptions of the application's functions. Each section of the document is listed below with a short summary.

- Overall Description — Describes the general factors that affect the product and its requirements.
- Specific Requirements — Details the software requirements.
- Classification of Functional Requirements — Lists the requirements and their type.

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

2. Overall Description

The Overall Description section of this Software Requirements Specification explains the general factors that affect the product and its requirements. This section will provide background for said requirements, which are further defined in thorough detail in Section 3.

2.1 Product Perspective

The infix calculator is a standalone software application designed to evaluate mathematical expressions provided by users. It does not have direct dependencies on other software systems or applications.

2.1.1 System Interfaces

The calculator interacts with the user through a text-based interface, taking input expressions and providing computed results. It may also use standard input and output for interaction.

2.1.2 User Interfaces

The user interacts with the calculator through a command-line interface (CLI). Users enter mathematical expressions as text, and the calculator displays results on the command line.

2.1.3 Hardware Interfaces

The calculator operates on a standard computer or computing device with no specific hardware interface requirements.

2.1.4 Software Interfaces

The calculator is implemented in C++ and may use standard libraries for input/output and mathematical operations. It interfaces with the C++ standard libraries as well as any custom data structure implementations.

2.1.5 Communication Interfaces

The calculator does not have external communication interfaces or network connectivity requirements.

2.1.6 Memory Constraints

The calculator should be memory-efficient and not consume excessive memory. However, specific memory constraints are not defined.

2.1.7 Operations

The calculator performs various mathematical operations, including addition, subtraction, multiplication, division, modulus (modulo), exponentiation, and handling of parentheses. It also handles error conditions and input validation.

2.2 Product functions

The primary function of the calculator is to evaluate infix mathematical expressions provided by users. It must correctly handle operators, precedence, and parentheses while producing accurate results. Optionally, it may support exponentiation using either the "^" or "***" operators.

2.3 User characteristics

The calculator is designed for users who need to perform arithmetic and algebraic calculations. Users should have a basic understanding of mathematical expressions and operators.

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

2.4 Constraints

- Must use some sort of implementation of data structures.
- Must be able to handle parentheses.
- Must be able to handle exponents via “^” and “**” (optional).
- Must handle all operators including modulo.
- Must handle errors correctly.
- Must be programmed using C++.

2.5 Assumptions and Dependencies

- The calculator assumes that user input will be in a valid format, adhering to the specified constraints.
- It depends on the correctness of the C++ compiler and standard libraries.

2.6 Requirements Subsets

The requirements subsets include specific functionalities and constraints, such as:

- Handling parentheses correctly.
- Supporting exponentiation using “^” or “**” (optional).
- Correctly managing operator precedence.
- Implementing robust error handling.
- Using C++ for implementation.

3. Specific Requirements

The Specific Requirements section of this Software Requirements Specification (SRS) includes all software requirements, explained in great detail to ensure designers can design a system to satisfy these requirements.

3.1 Functionality

- Addition +
- Subtraction -
- Multiplication *
- Division /
- Exponentiation ^
- Modulo %
- Parentheses ()

3.1.1 Addition

Add the values to the left and the right of the “+” symbol. Ensures that addition occurs in the equation in accordance with PEMDAS.

3.1.2 Subtraction

Subtract the values to the left and the right of the “-” symbol. Ensures that subtraction occurs in the equation in accordance with PEMDAS.

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

3.1.3 *Multiplication*

Multiply the values to the left and the right of the “*” symbol. Ensures that multiplication occurs in the equation in accordance with PEMDAS.

3.1.4 *Division*

Divide the values to the left and the right of the “/” symbol. Ensures that division occurs in the equation in accordance with PEMDAS.

3.1.5 *Exponentiation*

Multiply the number to the left of the “^” symbol by itself as many times as the number to the right of the “^” symbol. Ensures that exponentiation occurs in the equation in accordance with PEMDAS.

3.1.6 *Modulo*

Divides the number to the left of the “%” symbol by the number to the right and returns the remainder. Ensures that modulo occurs in the equation in accordance with PEMDAS.

3.1.7 *Parentheses*

Prioritizes the calculation to the right of the “(” symbol and to the left of the corresponding “)” symbol. Ensures that parentheses occur in the equation in accordance with PEMDAS.

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

3.2 Use-Case Specifications

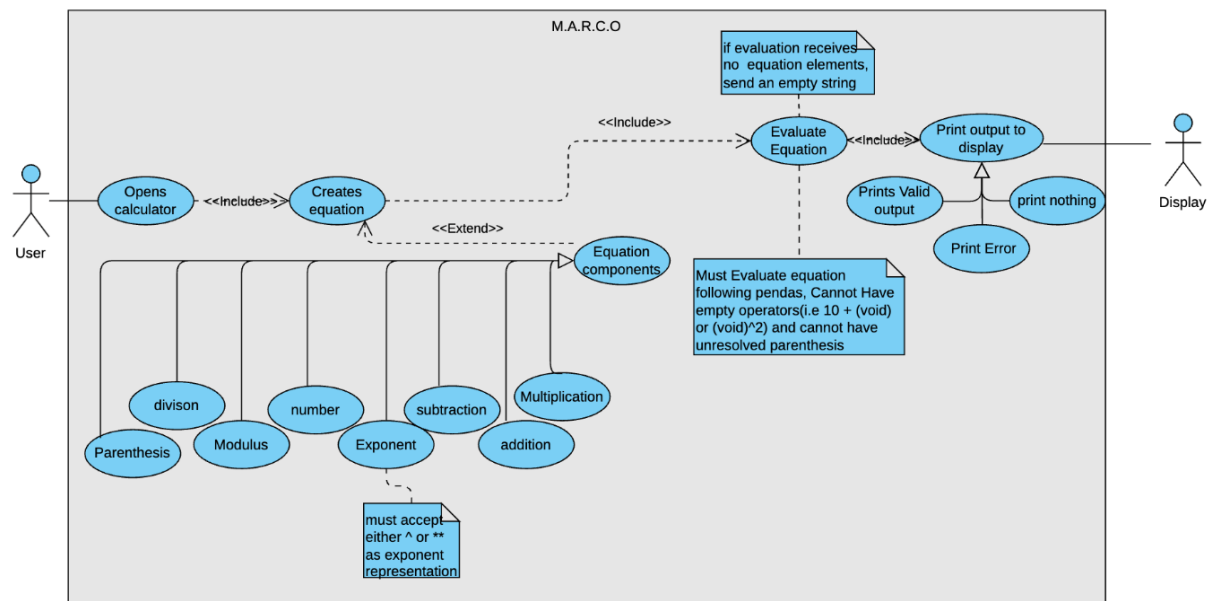


Photo of Use Case Diagram in Visual Paradigm.

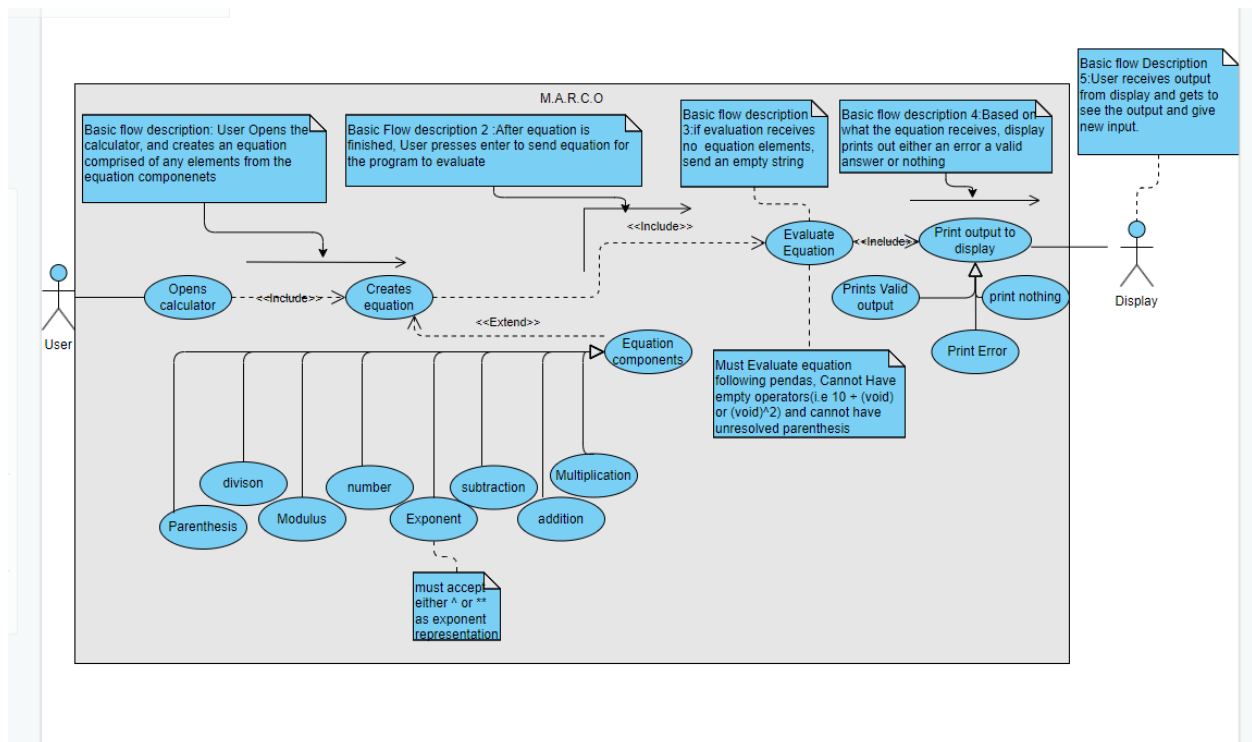


Photo of Use Case Diagram in Visual Paradigm.

Please refer to Appendix 1 at the end of this document.

M.A.R.C.O.	Version: 1.1
Software Requirements Specifications	Date: 15/10/23
PROJ2023-SDP-001	

3.3 Supplementary Requirements

- Must be written in C++.
- Must be accessible from the cmd prompt.
- Calculations must be returned within 5 seconds.
- User entered text should not be able to be turned into an executable.
- Calculations must be correct every time.
- Code should be easily repairable and easily scalable.

4. Classification of Functional Requirements

Functionality	Type
Parse Expressions	Essential
Operation Support (+, -, *, /, ^, %)	Essential
Operation Support (**)	Desirable
Handle Parentheses	Essential
Recognize Numeric Constants	Essential

5. Appendices

Appendix A: M.A.R.C.O. Use-Case Diagram

This diagram should be considered a part of the requirements.

https://github.com/pashiav/348_project/blob/main/M.A.R.C.O_Use-case_Diagram_SRS-Apdx1.pdf.pdf