Group 4

Arithmetic Expression Evaluator
Software Requirements Specifications

Version 1.0

Revision History

Date	Version	Description	Author
9/25/2024	0.0	Initial meeting, discussion of meeting times, and assigning roles for the project	Alexander Carrillo
10/16/2024	1.0	Software requirements discussion, reviewing rough code, talking about next steps	Alexander Carrillo

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 5
   - 2.5 Assumptions and dependencies 5
   - 2.6 Requirements subsets 5

3. Specific Requirements 5
   - 3.1 Functionality 5
     - 3.1.1 <Functional Requirement One> 6
   - 3.2 Use-Case Specifications 6
   - 3.3 Supplementary Requirements 6

4. Classification of Functional Requirements 6

5. Appendices 6

Software Requirements Specifications

1. Introduction

The Arithmetic Expression Evaluator aims to be a user interface calculator that can handle operators for addition, subtraction, multiplication, division, exponents, modulus, and order of operations. Users will be able to input an equation using understandable arithmetic and then receive an output. PEMDAS will be used in the document and further ones to describe the process of the order of operations.

1.1 Purpose

This software requirements specifications document will show the requirements, functional and nonfunctional, and constraints for the project.

1.2 Scope

*[A brief description of the software application that the **SRS** applies to, the feature or other subsystem grouping, what Use-Case model(s) it is associated with, and anything else that is affected or influenced by this document.]*

1.3 Definitions, Acronyms, and Abbreviations

PEMDAS: describes the process of the order of operations, parentheses, exponents, multiplication, division, addition, and subtraction. in that order

Operators: the symbols that the program will read from the user input, +, -, *, **, /, %

1.4 References

"Other documents" refers to the previous software development plan, or project plan

in future documents it will refer to previous documents which can be found here (github link)

1.5 Overview

The rest of the document describes requirements, functionality, and use case specifications.

Overall Description

1.6 Product perspective

1.6.1 System Interfaces

1.6.1.1 Arithmetic calculator code
1.6.1.2 user interface code

1.6.2 User Interfaces

1.6.2.1 console input (x + y)

1.6.3 Hardware Interfaces

1.6.3.1 Computer

1.6.4 Software Interfaces

1.6.4.1 computer

1.6.5 Communication Interfaces

1.6.6 Memory Constraints

1.6.6.1

1.6.7 Operations

1.6.7.1 addition
1.6.7.2 subtraction
1.6.7.3 multiplication
1.6.7.4 division
1.6.7.5 exponents
1.6.7.6 modulation

1.7 Product functions

1.7.1 arithmetic operations
1.7.2 parentheses
1.7.3 PEMDAS

1.8 User characteristics

1.8.1

1.9 Constraints

1.9.1

1.10 Assumptions and dependencies

1.10.1

1.11 Requirements subsets

1.11.1

2. Specific Requirements

*[This section of the **SRS** contains all software requirements to a level of detail sufficient to enable designers to design a system to satisfy those requirements, and testers to test that the system satisfies those requirements. When using use-case modeling, these requirements are captured in the Use Cases and the applicable supplementary specifications. If use-case modeling is not used, the outline for supplementary specifications may be inserted directly into this section, as shown below.]*

2.1 Functionality

*[This section describes the functional requirements of the system for those requirements that are expressed in the natural language style. For many applications, this may constitute the bulk of the **SRS** package and thought should be given to the organization of this section. This section is typically organized by feature, but alternative organization methods may also be appropriate; for example, organization by user or organization by subsystem. Functional requirements may include feature sets, capabilities, and security.*

Where application development tools, such as requirements tools, modeling tools, and the like, are employed to capture the functionality, this section of the document would refer to the availability of that data, indicating the location and name of the tool used to capture the data.]

2.1.1 Operations

- able to recognize and then perform operations such as addition, subtraction, multiplication, division, modulation, and exponentes

2.1.2 User Interface

- User is able to input arithmetic problems such as 2 + 2 (most basic) or 2+4*2/((5-2)/4) (more advanced) and then be shown the correct output

2.1.3 Order of operations

- arithmetic problems are done in the PEMDAS order of operations, parentheses > exponent > multiplication & division > addition and subtraction

2.1.4 Exception handling

- If the user inputs something that can not be computed the program should not crash and should instead give an error message and the give the user another chance for an input

2.1.5 multiple kinds of parentheses

- Allows for the use of [] and {} as substitutes for parentheses

2.2 Use-Case Specifications

Basic Example: The user inputs 7+4, the system splits into two ints, and the operator reads and calls the addition function. the function returns the sum of 7 and 4, 11, then shows it in the console

Parentheses and Pemdas example: The user inputs 2+4*5+(2-1), the system breaks it into parts starting with what is in the parentheses, calls subtract, returns 1. Next is multiplication, call multiply, and returns 20. next, we just go from left to right, doing: 2+20 = 22, 22+ 1 =23. shows 23.

you could break down use cases to each operator (addition, multiplication, etc.) and combinations of them, the most basic use case for the calculator would be, “do an arithmetic problem”, then branching off into specifics from there.

2.3 Supplementary Requirements

[Supplementary Specifications capture other requirements, e.g., non-functional requirements and development constraints, that are not included in the use cases and non-functional requirements.]

3. Classification of Functional Requirements

Functionality	Type
Operators (add, sub, mult, div, mod, ex)	Essential
User interfacing	Essential

Order of Operations (PEMDAS)	Essential
Multiple types of parentheses ([], {}, etc.)	Desirable
Exception handling (dividing by zero, unmatched parentheses)	Essential

4. Appendices

*[When appendices are included, the **SRS** should explicitly state whether or not the appendices are to be considered part of the requirements]*