EvalEx

**Software Development Plan**

**Version 1.0**

**Revision History**

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 24/09/23 | 1.0 | Initial addition of project details | Will Whitehead, Joshua Lee, Joe Hotze, Max Biundo, Tommy Lam |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

[**1.**](#_vx1227) **Introduction 4**

[*1.1*](#_3fwokq0) *Purpose 4*

[*1.2*](#_1v1yuxt) *Scope 4*

[*1.3*](#_4f1mdlm) *Definitions, Acronyms, and Abbreviations 4*

[*1.4*](#_2u6wntf) *References 4*

[*1.5*](#_19c6y18) *Overview 5*

[**2.**](#_3tbugp1) **Project Overview 5**

[*2.1*](#_28h4qwu) *Project Purpose, Scope, and Objectives 5*

[*2.2*](#_nmf14n) *Assumptions and Constraints 5*

[*2.3*](#_37m2jsg) *Project Deliverables 5*

[*2.4*](#_1mrcu09) *Evolution of the Software Development Plan 5*

[**3.**](#_46r0co2) **Project Organization 5**

[*3.1*](#_2lwamvv) *Organizational Structure 5*

[*3.2*](#_111kx3o) *External Interfaces 6*

[*3.3*](#_3l18frh) *Roles and Responsibilities 6*

[**4.**](#_206ipza) **Management Process 6**

[*4.1*](#_4k668n3) *Project Estimates 6*

[*4.2*](#_2zbgiuw) *Project Plan 6*

[*4.3*](#_1egqt2p) *Project Monitoring and Control 7*

[*4.4*](#_qsh70q) *Requirements Management 7*

[*4.5*](#_49x2ik5) *Quality Control 7*

[*4.6*](#_147n2zr) *Reporting and Measurement 7*

[*4.7*](#_ihv636) *Risk Management 8*

[*4.8*](#_1hmsyys) *Configuration Management 8*

[**5.**](#_2grqrue) **Annexes 8**

**Software Development Plan**

# **Introduction**

## **Purpose**

The purpose of the *Software Development Plan* is to gather all information necessary to control the project. It describes the approach to the development of the software and is the top-level plan generated and used by managers to direct the development effort.

The following people use the *Software Development Plan*:

* The **Project Manager** uses it to plan the project schedule and resource needs, and to track progress against the schedule.
* The **Risk Manager** operatoruses it to see what risks are seen and how to take different approaches based on the SDP.
* The **Quality Control** operator uses it to understand what requirements are needed and how best to implement them.
* The **Configuration Management** operator uses it to ensure the systems are maintained in a suitable state.
* The **Programmers** will write the code for the project.

## **Scope**

This *Software Development Plan* describes the overall plan to be used by the EvalEx project, including deployment of the product. The details of the individual iterations will be described in the Iteration Plans.

## **Definitions, Acronyms, and Abbreviations**

N/A (Will likely change upon further development)

## **References**

*For the* ***Software Development Plan****, the list of referenced artifacts includes:*

N/A (Will change upon further development)

## **Overview**

This *Software Development Plan* contains the following information:

* Project Overview: Provides a description of the project's purpose, scope, and objectives.  It also defines the deliverables that the project is expected to deliver.
* Project Organization: Describes the organizational structure of the project team.
* Management Process: Explains the estimated cost and schedule, defines the major phases and milestones for the project, and describes how the project will be monitored.
* Applicable Plans and Guidelines: Provide an overview of the software development process, including methods, tools and techniques to be followed.

# **Project Overview**

## **Project Purpose, Scope, and Objectives**

The purpose of this project is to create a calculator that can do simple PEMDAS operations. The project is expected to give a valid output for the operations +,-,\*,/,(),and ^ with valid number input.

## **Assumptions and Constraints**

There are a few constraints on the project. First, the project has to be in C / C++. We are limited to six group members at the very most. We can meet as many times a week as needed, but we have until the end of the semester, almost the middle of december, to completely finish the project

## **Project Deliverables**

* Requirements: 10/29
* Design Specs: 11/5
* Implementation: 11/16
* Test Cases: 12/1
* Code: 12/7

Deliverables for each project phase are identified in the Development Case. Deliverables are delivered towards the end of the iteration, as specified in section *4.2.4 Project Schedule*.

## **Evolution of the Software Development Plan**

The *Software Development Plan* will be revised prior to the start of each Iteration phase.

# **Project Organization**

## **Organizational Structure**

* Project Leader: Max Biundo
* Risk Management: Joe Hotze
* Quality Control: Will Whitehead
* Configuration Management: Joshua Lee
* Programmer: Gunther Luechtefeld, Tommy Lam

## **External Interfaces**

*N/A*

## **Roles and Responsibilities**

| **Person** | **Unified Process for EDUcation Role** |
| --- | --- |
| Max Biundo | Project Leader - Guides the rest of the group through the project. In charge of the project and its direction. |
| Joe Hotze | Risk Management - Finds and eliminates any and all risks that could arise throughout the course of the development. |
| Will Whitehead | Quality Control - Searches for bugs or possible errors in the code to fix before they become significant issues. |
| Joshua Lee | Configuration Management - Make sure project artifacts are sufficiently named and marked. |
| Tommy Lam | Programmer - Write the code for the project. |
| Gunther Leuchtefeld | Programmer - Write the code for the project. |

Anyone on the project can perform Any Role activities.

# **Management Process**

## **Project Estimates**

*N/A*

## **Project Plan**

* Project Management Plan: 9/24/2023
* Project Requirements: 9/30/2023
* Project Architecture and Design: 10/26/2023
* Project Implementation: 11/02/2023
* Project Test Cases: 11/16/2023
* Project User Manual: 11/30/2023
* Project Implementation Due: Updated project management. plan, requirements, design, test cases, code, user manual: 12/07/2023

### **Phase Plan**

*N/A*

### **Iteration Objectives**

* Iteration 1: Implement ‘+,-,/,\*’ operator
* Iteration 2: Implement operator precedence
* Iteration 3: Implement parenthesis handling
* Iteration 4: Implement proper UI
* Iteration 5: Implement error handling

### **Releases**

N/A (Will be updated further in development)

### **Project Schedule**

* Iteration 1: 11/24
* Iteration 2: 11/27
* Iteration 3: 12/1
* Iteration 4: 12/4
* Iteration 5: 12/7

### **Project Resourcing**

*N/A*

## **Project Monitoring and Control**

* Requirements Management: N/A
* Quality Control: Programmers and others will test code after and throughout each iteration in order to ensure quality.
* Reporting and Measurement: N/A
* Risk Management: Steps will be taken in order to minimize primary risk of failing to attend meetings.
* Configuration Management: Team process, prevents unnecessary and bloated code.

## **Requirements Management**

N/A

## **Quality Control**

Each version will be tested in order to ensure it meets the requirements of the rubric. Code will be written and reviewed separately after each iteration by different programmers to guarantee successful program operation.

## **Reporting and Measurement**

N/A

## **Risk Management**

The most apparent risk throughout development will be the potential for last-minute engagements interrupting a plan to meet and work on a project. This can be minimized by meeting early and spreading out workloads over several meetings.

## **Configuration Management**

After each iteration of code is created, other programmers will look at the code put forward and critique it for any potential changes that need to be made in order to ensure a sleek, high speed program with minimal “fluff” code.

# **Annexes**

*N/A*