Arithmetic Expression Evaluator

Software Development Plan

Version <1.0>

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| <dd/mmm/yy> | <x.x> | <details> | <name> |
| 18/09/2024 | 1.0 | Initial editing of the document to fill out all required sections and create a meeting log. | Entire team |
| 24/09/2024 | 1.1 | Edited out blue text, yellow and green highlights, added gantt chart | Maxwell Phachanla |
|  |  |  |  |

Table of Contents *[keep this; say N/A when inapplicable]*

[**1.**](#_heading=h.30j0zll) **Introduction 4**

[*1.1*](#_heading=h.1fob9te) *Purpose 4*

[*1.2*](#_heading=h.3znysh7) *Scope 4*

[*1.3*](#_heading=h.2et92p0) *Definitions, Acronyms, and Abbreviations 4*

[*1.4*](#_heading=h.tyjcwt) *References 4*

[*1.5*](#_heading=h.3dy6vkm) *Overview 5*

[**2.**](#_heading=h.1t3h5sf) **Project Overview 5**

[*2.1*](#_heading=h.4d34og8) *Project Purpose, Scope, and Objectives 5*

[*2.2*](#_heading=h.2s8eyo1) *Assumptions and Constraints 5*

[*2.3*](#_heading=h.17dp8vu) *Project Deliverables 5*

[*2.4*](#_heading=h.3rdcrjn) *Evolution of the Software Development Plan 5*

[**3.**](#_heading=h.26in1rg) **Project Organization 5**

[*3.1*](#_heading=h.lnxbz9) *Organizational Structure 5*

[*3.2*](#_heading=h.35nkun2) *External Interfaces 6*

[*3.3*](#_heading=h.1ksv4uv) *Roles and Responsibilities 6*

[**4.**](#_heading=h.44sinio) **Management Process 6**

[*4.1*](#_heading=h.2jxsxqh) *Project Estimates 6*

[*4.2*](#_heading=h.z337ya) *Project Plan 6*

[*4.3*](#_heading=h.3whwml4) *Project Monitoring and Control 7*

[*4.4*](#_heading=h.qsh70q) *Requirements Management 7*

[*4.5*](#_heading=h.49x2ik5) *Quality Control 7*

[*4.6*](#_heading=h.147n2zr) *Reporting and Measurement 7*

[*4.7*](#_heading=h.ihv636) *Risk Management 8*

[*4.8*](#_heading=h.1hmsyys) *Configuration Management 8*

[**5.**](#_heading=h.2grqrue) **Annexes 8**

Software Development Plan

# 

# Introduction

This Software Development Plan is to provide an overview and key details about the project, who the members are, the purpose of the project, and keep track of important info such as dates for project revisions.

## Purpose

The purpose of the *Software Development Plan* is to detail all information related to the project. This includes what the aim of the project is and the approach to the development of the software. This document is the top-level plan generated by all the team members to explain the steps of 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 ensure the team is making progress at the right pace.
* **Project team members** use it to understand their objectives and deadlines, as well as the overview of the project.

## Scope

This *Software Development Plan* describes the overall plan to be used by the Arithmetic Expression Evaluator project, including deployment of the product. It also describes the different roles of team members. The details of the individual iterations will be described in the Iteration Plans.  
The plans as outlined in this document are based upon the product requirements as defined in the *Vision Document*.

## Definitions, Acronyms, and Abbreviations

*+* (addition)

- (subtraction)

\* (multiplication)

/ (division)

% (modulo)

\*\* (exponentiation)

## References

* [*GitHub*](https://github.com/MinoMax0205/EECS-348-Project---Group-1) *repository*
* [*Meeting Log*](https://docs.google.com/document/d/1jiPPu5WAUli79O-ZVyhve-rKLh3cqSeR7KOyC5I2abE/edit)

## 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 object of this project is to build an arithmetic expression evaluator in C++. The program is meant to parse and calculate arithmetic expressions with operators and numeric constants. The program will take input from the user, parse it, and calculate a result according to the order of operations. The purpose of this project is to understand the development lifecycle, expose team members to integrating someone else’s work, and reinforce our understanding of parsing techniques and algorithm design.

## Assumptions and Constraints

The ability to do/learn C++

Schedule/meeting times being in the morning or night for attendance of all members

Project has to be coded in C++

Balancing our schedule for the project and other personal tasks

## Project Deliverables

Project Plan due 9/29

Project Requirements due 10/13

Project Architecture and Design due 11/10

Project Implementation due 12/12

Project Test Cases due 12/12

Project User Manual due 12/12

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

Megan Svoren is Technical Leader due to her in-depth knowledge of many program languages.

Maxwell Phachanla is the Assistant Team Administrator to make sure that meeting logs are recorded and kept.

Bryce Martin - Quality Assurance: Will make sure all documents and code is up to quality standard.

Luke Kounkel - Assistant Technical Manager: Will learn C++ and assist Technical leader in writing the project code.

Jacob Richards- Team administrator, make sure everything is filled out in the documents,meeting logs are tracked, scheduling, deadlines are met, turn ins are done.

## External Interfaces

The project will be graded either by the course’s professor, Hossein Saiedian, or one of the TAs. They will judge the quality of our teamwork and the project itself.

## Roles and Responsibilities

| **Person** | **Unified Process for EDUcation Role** |
| --- | --- |
| Maxwell Phachanla | Assistant Team Administrator / [m281p216@ku.edu](mailto:m281p216@ku.edu) / 7853424406  Availability: M 9:30 - 10:50 AM & 3:00 PM -, TT 9:30 - 12:30 AM, W 9:30 - 10:50 AM & 3:00 PM -, TH 9:30 - 10:50 AM & 5:30 PM -, F NA |
| Bryce Martin | Quality Assurance Engineer / 913-832-0292 / [brycemartin@ku.edu](mailto:brycemartin@ku.edu), |
| Luke Kounkel | Assistant Technical Manager/ [lukekounkel@gmail.com](mailto:lukekounkel@gmail.com)/ 9133374155  Availability: M 10am-12am, TT 10:30am-1pm, WF -12am |
| Noah Hamlet | Project Leader, 785-304-5867, [noahhamlet08@gmail.com](mailto:noahhamlet08@gmail.com), availability: M-F 9-11 a.m. |
| Megan Svoren | Technical Leader / 913-280-4760 / [megan.svoren@ku.edu](mailto:megan.svoren@ku.edu)  Availability: MWF 9-11am |
| Jacob Richards | Team Administrator 913-534-1139, [j140r614@ku.edu](mailto:j140r614@ku.edu), availability M-F 9-11am , weekends as needed |

Anyone on the project can perform [Any Role](about:blank) activities.

# Management Process

## Project Estimates

No estimated cost except for time and sleep.

## Project Plan

Complete Project Plan by 9/29

Complete Project Requirements by 10/13

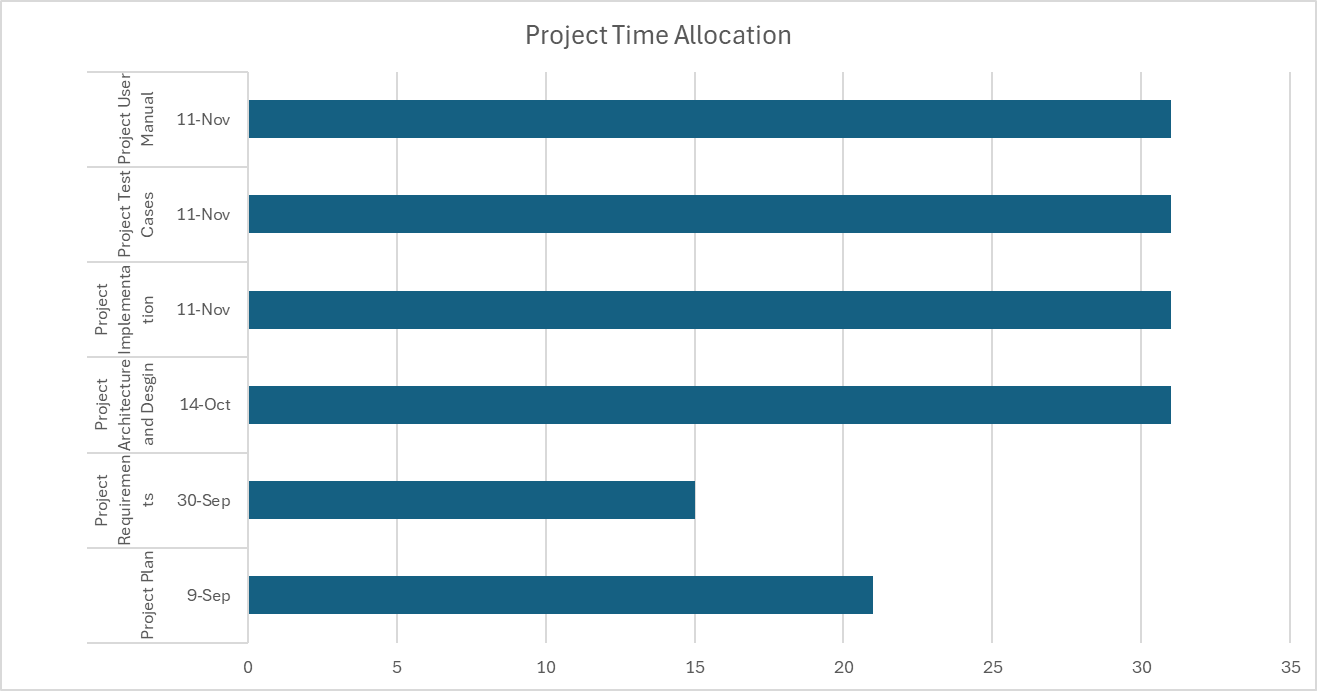
Complete Project Architecture and Design by 11/10

Complete Project Implementation by 12/12

Complete Project Test Cases by 12/12

Complete Project User Manual by 12/12

### Phase Plan

**

### Iteration Objectives

Project Plan

Project Requirements

Project Architecture and Design

Project Implementation

Project Test Cases

Project User Manual

### Releases

NA

Arithmetic expression evaluator beta

### Project Schedule

Project Plan due 9/29

Project Requirements due 10/13

Project Architecture and Design due 11/10

Project Implementation due 12/12

Project Test Cases due 12/12

Project User Manual due 12/12

### Project Resourcing

No special training is required, except for team members to learn C++ before development of the project beings (e.g. before 10/13)

## Project Monitoring and Control

VisualStudios will be providing most of the control mechanisms through error detections and coded constraints/parameters. The Quality Assurance Manager and the other members will regularly check the project to make sure that it is on schedule to be completed.

The team will ensure quality control by meticulously reviewing each other’s work. We will utilize Visual Studio, an effective source code editor equipped with robust error detection and debugging capabilities, to maintain accuracy. Regular testing of the code will be performed to ensure consistency and reliability.

For managing version control, we will use GitHub, which allows us to effortlessly track code modifications and support collaborative development. Additionally, we will maintain detailed documentation, covering project plans, meeting notes, ideas, and changes, to ensure clarity and coherence throughout the project’s duration.

To minimize risk, we will implement a multi-step testing process alongside comprehensive code commenting. This approach ensures that potential issues are identified early and that the code is well understood by all team members

## **Requirements Management**

The requirements for this system are captured in the Vision document. Requested changes to requirements are captured in Change Requests, and are approved as part of the Configuration Management process.

## **Quality Control**

Defects will be recorded and tracked as Change Requests, and defect metrics will be gathered (see Reporting and Measurement below).

All deliverables are required to go through the appropriate review process, as described in the Development Case. The review is required to ensure that each deliverable is of acceptable quality, using guidelines and checklists.

Any defects found during review which are not corrected prior to releasing for integration must be captured as Change Requests so that they are not forgotten.

## **Reporting and Measurement**

Updated schedule estimates, and metrics summary reports, will be generated at the end of each iteration.

The Minimal Set of Metrics, as described in the RUP Guidelines: Metrics will be gathered on a weekly basis. These include:

Earned value for completed tasks. This is used to re-estimate the schedule and budget for the remainder of the project, and/or to identify need for scope changes.

Total defects open and closed – shown as a trend graph. This is used to help estimate the effort remaining to correct defects.

Acceptance test cases passing – shown as a trend graph. This is used to demonstrate progress to stakeholders.

*Refer to the Project Measurements Document (AAA-BBB-X.Y.doc) for detailed information.*

## **Risk Management**

Risks will be identified in the Inception Phase using the steps identified in the RUP for Small Projects activity “Identify and Assess Risks”. Project risk is evaluated at least once per iteration and documented in this table.

*Refer to the Risk List Document (CCC-DDD-X.Y.doc) for detailed information.*

## **Configuration Management**

Appropriate tools will be selected which provide a database of Change Requests and a controlled versioned repository of project artifacts.

All source code, test scripts, and data files are included in baselines. Documentation related to the source code is also included in the baseline, such as design documentation. All customer deliverable artifacts are included in the final baseline of the iteration, including executables.

The Change Requests are reviewed and approved by one member of the project, the Change Control Manager role.

*Refer to the Configuration Management Plan (EEE-FFF-X.Y.doc) for detailed information.*

# 

# Annexes

The project will follow the UPEDU process.

Other applicable process plans are listed in the references section, including Programming Guidelines.