LogiCalc
Software Development Plan
Version 1.0

Revision History

Date	Version	Description	Author
02/20/2024	0.1	Document initiation	Yara, Olivia
02/25/2024	1.0	First version completed	Yara, Olivia
03/24/2024	1.1	Revision 1	Yara

Table of Contents

Software Development Plan

1. Introduction

This document will guide the development team through the entire lifecycle of the project, from the initial planning and design phases to implementation, testing, and deployment. It will define the scope of the Boolean Expression Calculator, outline the key features and functionalities, detail the development methodology, and establish the testing and quality assurance processes to ensure a reliable and high-performance product.

1.1 Purpose

This Software Development Plan gathers and organizes the information necessary to maintain the project's timeline. It describes the approach to the development of the software and provides a timeline to be followed by the project team members.

The following people use the *Software Development Plan*:

- The **team leader** uses it to plan the project schedule and resource needs, and to track progress against the schedule.

- **Project team members** use it to understand what they need to do, when they need to do it, and what other activities they are dependent upon.

- **Upper management** uses it to maintain the project's timeline and requirements and communicate with the team leader about any changes.

1.2 Scope

This *Software Development Plan* describes the plan to be used by the Boolean Expression Calculator project, including completion of the product. The details of the individual iterations will be described in the Iteration Plans. The plans outlined in this document are based on the product requirements defined in the Software Requirements Specifications. This product is a standalone Boolean Expression Calculator with a command line interface (CLI).

1.3 Definitions, Acronyms, and Abbreviations

CLI: command line interface

GUI: graphical user interface

See the Project Glossary.

1.4 References

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

- Team Roles

- Software Requirements Specification

- Iteration Plans

- Software Architecture

- Test Cases

- User Manual

1.5 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.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

The purpose of the Boolean Expression Calculator project is to develop a specialized software tool with a command line interface for the evaluation and manipulation of Boolean expressions. The project's scope includes the design, development, testing, and deployment of a command line-based Boolean Expression Calculator. The application will provide a comprehensive and user-friendly solution that empowers users to work with Boolean expressions seamlessly through a command line interface.

2.2 Assumptions and Constraints

Platform Compatibility: The Boolean Expression Calculator will be developed to run on major operating systems such as Windows, Linux, and macOS.

Resource Limitations: The application should be designed to operate efficiently within reasonable memory and processing power constraints.

Command Line Only: The primary interface for this calculator will be command line-based, with considerations for potential future graphical user interface (GUI) enhancements.

2.3 Project Deliverables

The Boolean Expression Calculator project will produce a set of key deliverables, each playing a crucial role in the development, testing, and deployment of the software. Deliverables are delivered according to section 4.2.4 Project Schedule. The deliverables are as follows:

- Software Requirements Specification (FRD and NRD)
- Software Architecture (SDD and UI)
- Source Code and Executable Application
- Test Cases, Data and Reports
- User Manual
- Managements Plans and Reports
- Software

2.4 Evolution of the Software Development Plan

The *Software Development Plan* will be revised prior to the start of each Iteration phase. After the completion of each Iteration phase, it will be revised again to maintain consistency within the project.

3.   Project Organization

3.1   Organizational Structure

Six team members are completing this project, each maintaining a specific role with individual requirements. The team members communicate frequently, making sure to maintain consistency within the project. The team leader is the primary point of communication between the team and upper management.

3.2   Roles and Responsibilities

Requirements for each role and contact information are specified in the Team Roles Document.

Person	Unified Process for EDUcation Role
Yara Al-Shorman	Team Leader
Olivia Blankenship	Assistant Team Leader
Mariam Oraby	Product Owner
Aniketh Aatipamula	Technical Lead
Sam Suggs	Design & Testing Engineer
Zachary Schneider	Full-Time Software Engineer

4.   Management Process

4.1   Project Estimates

It is estimated that this project will have its final release in April 2024, contingent upon the client's needs and specifications.

No budget will be allocated towards this project.

4.2   Project Plan

4.2.1   *Iteration Objectives*

Sprint	Objectives
1	Complete software requirements spec document
2-3	Work on SRS document
4	Development, work on architecture docucment
5	Development, complete user manual document and complete test cases document
6	Testing & Debugging, deliver final software product release, complete and submit all required documents

Releases

Release	Version	Description
N/A	N/A	N/A

Project Schedule

Date	Goal	Sprint	Description
Feb 17, 2024	Group formation	Pre-sprint	Form team, designate roles
Feb 25, 2024	Project initiation	1	Complete project plan document and start on software requirements document
March 3, 2024	Preparation	2	SRS writing
Mar 11-15, 2024	Spring Break	-	Project Paused
March 17, 2024	Preparation	3	SRS writing
March 31, 2024	Development	4	Start development
April 14, 2024	Development	5	Continue and finalize development
April 28, 2024	Final Software Release	6	Testing & Debugging. Final release of the product
May 5, 2024	Final Project Delivery	Post-sprint	Deliver final product alongside project documentation

4.3   Project Monitoring and Control

4.4   Quality Control

It is the responsibility of the Product Owner to ensure the promised quality of the software product is delivered. Defect and change requests shall be documented through the Issues tab in the GitHub repository linked here: https://github.com/aaatipamula/bool_expr_calc

Merge requests to the GitHub repository must go through a peer review process to minimize errors and mistakes.

There shall be small-scale testing performed at the end of each sprint by the Product Owner and the Technical Lead, and an official and final testing and debugging sprint (outlined in sprint 6) completely allocated towards ensuring the quality of the final software product.

Iteration (sprint) goals shall be outlined in the Iterations Plan document, software requirements shall be identified and specified in the Software Requirements Specification document.

More details on Quality Control will be added later.

4.5   Risk Management

LogiCalc is a low-risk product, there have been no identified risks associated with using it.

4.6   Configuration Management

- Quality Analysis and Assurance is the responsibility of the Product Owner and they shall be the one to communicate any software defects or issues to the development team.
- They shall use GitHub and the regularly scheduled team meetings to communicate change requests, which will also be documented in the Iterations Plan document in more detail and also to assign those tasks to specific members of the development team to complete over the period of the upcoming sprint.
- Backups can be made by rolling back on software releases using the GitHub repository.
- Development tasks are assigned at the beginning of each spring and tasks are not to be added to the sprint after it started unless deemed necessary and urgent by the Product Owner and Technical Lead.
- Documents are named using the name specified by UPEDU format and also using a document identifier such A0, which refers to this document.

5.   Annexes

The project will follow the UPEDU process.

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