<Calculator>

Version <1.0>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| <12/09/23> | <1.0> | <Initial additions to the outline> | <Xavier Ruyle> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1. Introduction 4](#_Toc11132094)

[1.1 Purpose 4](#_Toc11132095)

[1.2 Scope 4](#_Toc11132096)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc11132097)

[1.4 References 4](#_Toc11132098)

[1.5 Overview 5](#_Toc11132099)

[2. Project Overview 5](#_Toc11132100)

[2.1 Project Purpose, Scope, and Objectives 5](#_Toc11132101)

[2.2 Assumptions and Constraints 5](#_Toc11132102)

[2.3 Project Deliverables 5](#_Toc11132103)

[2.4 Evolution of the Software Development Plan 5](#_Toc11132104)

[3. Project Organization 5](#_Toc11132105)

[3.1 Organizational Structure 5](#_Toc11132106)

[3.2 External Interfaces 6](#_Toc11132107)

[3.3 Roles and Responsibilities 6](#_Toc11132108)

[4. Management Process 6](#_Toc11132109)

[4.1 Project Estimates 6](#_Toc11132110)

[4.2 Project Plan 6](#_Toc11132111)

[4.3 Project Monitoring and Control 7](#_Toc11132112)

[4.4 Requirements Management 7](#_Toc11132113)

[4.5 Quality Control 7](#_Toc11132114)

[4.6 Reporting and Measurement 7](#_Toc11132115)

[4.7 Risk Management 8](#_Toc11132116)

[4.8 Configuration Management 8](#_Toc11132117)

[5. Annexes 8](#_Toc11132118)

# 

# Introduction

A calculator is essential for any task that involves arithmetic. It is the goal of this development team to create a calculator that is able to handle basic expressions. The team will utilize the SYA (Shunting Yard Algorithm) to convert an arithmetical expression passed in by a human to RPN (Reverse Polish Notation) which is also understood as PN (Postfix Notation). The PN will then be sent through a stack, where the expression will be calculated while following correct PEMDAS rules.

## Purpose

The purpose of the Software Development Plan is to layout basic information that is vital to the project. It will guide the team members to correctly carry out the necessary tasks to create a basic working calculator. The plan will not only deal with technical issues relating to the creation of the calculator but also deal with the planning process and resources needed in order to successfully complete the project.

The following people use the Software Development Plan:

* **Project team members** use it to obtain a grasp on the algorithms and data structures that will be used to create the calculator
* The **project manager** will use it to make sure deadlines and requirements are met which encompass the entire project.

## Scope

This Software Development plan creates the framework for the development and testing process for the creation of a basic calculator. It contains the details regarding the plan to implement and code the calculator as well as the necessary dependencies and knowledge needed to create it.

## Definitions, Acronyms, and Abbreviations

See the Project Glossary.

## References

**Should place in appendix, will do that later.**

- Shunting Yard Algorithm Source: <https://www.engr.mun.ca/~theo/Misc/exp_parsing.htm>

[This subsection provides a complete list of all documents referenced elsewhere in the **Software Development Plan**. Identify each document by title, report number if applicable, date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.

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

* Iteration Plans
* Development Case
* Vision
* Glossary
* Any other supporting plans or documentation.

## Overview

[This subsection describes what the rest of the **Software Development Plan** contains and explains how the document is organized. The text below is provided as an example.]

This *Software Development Plan* contains the following information:

Project Overview  — Gives an overview of the calculator project’s purpose, scope, and goals.  It also defines the features and functions that calculator will have.

Project Organization  — Gives a description of the team’s organizational practices and makeup.

Management Process  — Gives the time schedule for different deliverables, as well as the cost and different goals for the project as well as how the project’s progression will be analyzed.

Applicable Plans and Guidelines — Gives an overview of the project’s software development process, including the algorithms, tools, and best practices it will follow.

# Project Overview

## Project Purpose, Scope, and Objectives

[A brief description of the purpose and objectives of this project and a brief description of what deliverables the project is expected to deliver.]

## Assumptions and Constraints

[A list of assumptions that this plan is based and any constraints, for example. staff, equipment, schedule, that apply to the project.]

## Project Deliverables

[A list of the artifacts to be created during the project, including target delivery dates. The text below is provided as an example.]

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

[A table of proposed versions of the **Software Development Plan**, and the criteria for the unscheduled revision and reissue of this plan. The text below is provided as an example.]

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

# Project Organization

## Organizational Structure

[Describe the organizational structure of the project team, including management and other review authorities.]

## External Interfaces

[Describe how the project interfaces with external groups. For each external group, identify the internal and external contact names. This should include responsibilities related to deployment and acceptance of the product.]

## Roles and Responsibilities

[Identify the project organizational units that will be responsible for each of the disciplines, workflow details, and supporting processes. The text below is provided as an example.]

|  |  |
| --- | --- |
| **Person** | **Unified Process for EDUcation Role** |
|  |  |
|  |  |

Anyone on the project can perform [Any Role](..\..\..\process\workers\wk_any.htm) activities.

# Management Process

## Project Estimates

[Provide the estimated cost and schedule for the project, as well as the basis for those estimates, and the points and circumstances in the project when re-estimation will occur.]

## Project Plan

[This section contains the schedule and resources for the project.]

### Phase Plan

[Include the following:

 a Gantt chart showing the allocation of time to the project phases (Not necessarily detailed to the activity level; this type of Gantt Chart is providing along with the Iteration Plans themselves; Provide an Overview of the project Timeline with the major miles stones]

 identify **major milestones** with their achievement criteria

Define any important release points and demos.]

[If available, refer to the related **Iteration Plan Documents** for more details]

### Iteration Objectives

[Briefly list the objectives to be accomplished for each of the iterations and Refer to the related **Iteration Plan Documents** for more details.]

### Releases

[A brief description of each software release and whether it’s demo, beta, and so on.]

### Project Schedule

[Diagrams or tables showing target dates for completion of iterations and phases, release points, demos, and other milestones.]

### Project Resourcing

 [Identify the numbers and type of staff required here, including any special skills or experience, scheduled by project phase or iteration.

List any special training project team members will require, with target dates for when this training should be completed.]

## Project Monitoring and Control

 [The following is a checklist of items to consider:

* Requirements Management: Specify the information and control mechanisms which will be collected and used for measuring, reporting, and controlling changes to the product requirements.
* Quality Control: Describe the timing and methods to be used to control the quality of the project deliverables and how to take corrective action when required. Include techniques, metrics, criteria, and procedures used for evaluation— this will include walkthroughs, inspections, and reviews. Note that this is in addition to the Test Plan, which is not enclosed in the Software Development Plan.
* Reporting and Measurement: Describe reports to be generated. Specify which metrics should be collected and why. **OR** if available, refer to the **Project Measurements and Project Measurements** document
* Risk Management: Describe the approach that will be used to identify, analyze, prioritize, monitor and mitigate risks. If available, refer to the **Risk List** document.
* Configuration Management: Describe the process by which problems and changes are submitted, reviewed, and dispositioned. Describe how project or product artifacts are to be named, marked, and numbered, including system software, plans, models, components, test software, results and data, executables, and so on. Describe retention policies, and the back-up, disaster, and recovery plans. **OR** if Available, Refer to the **Configuration Management Plan** document

The text that follows is provided as an example.]

## **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 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

[Additional material of use to the reader of the **Software Development Plan**. Reference or include any project technical standards and plans which apply to this project. This typically includes the Programming Guidelines, Design Guidelines, and other process guidelines. The text that follows is provided as an example.]

The project will follow the UPEDU process.

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