<BATT5>

Software Configuration Management Plan

Version <1.0>

9/26/19

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | 0.1 |
| Current Release: | 1.0 |
| Indicator of Last Page in Document: | & |
| Date of Last Review: | 9/24/19 |
| Date of Next Review: | 9/26/19 |
| Target Date for Next Update: | 9/26/19 |

Distribution List

This following list of people shall receive a copy of this document every time a new version of this document becomes available:

Guidance Team Members:

Dr. Gates

Dr. Salamah

Dr. Roach

Elsa Tai Ramirez

Jake Lasley

Customers: Dr. Jaime C. Acosta

Dr. Oscar A. Perez

Vincent Fonseca

Herandy Denisse Vazquez

Baltazar Santaella

Florencia Larsen

Juan Ulloa

Jesus Martinez

Software Team Members:

Adal Rivas

Alain Sanchez

Andrea Labrado

Mark Nunez

Juan Gaucin

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 0.1 | 9/5 | Team 5 | Started draft on sections 2, 3.2 and 3.3 |
| 0.2 | 9/24 | Team 5 | Completed Section 1,2,3 |
| 0.3 | 9/25 | Team 5 | Completed Section 4 |
| 1.0 | 9/26 | Team 5 | Complete/Revise Document |

Table of Contents

[Document Control ii](#_Toc20402249)

[Approval ii](#_Toc20402250)

[Document Change Control ii](#_Toc20402251)

[Distribution List ii](#_Toc20402252)

[Change Summary ii](#_Toc20402253)

[1. Introduction 1](#_Toc20402254)

[1.1. References 1](#_Toc20402255)

[2. Software Configuration Identification 2](#_Toc20402256)

[2.1. Software Configuration Item Identification 2](#_Toc20402257)

[2.2. Labeling Schema 2](#_Toc20402258)

[2.3. Directory Structure 3](#_Toc20402259)

[2.4. Back-Up Plan 3](#_Toc20402260)

[2.5. Organizational Process 3](#_Toc20402261)

[3. Software Configuration Item Organization 4](#_Toc20402262)

[3.1. Documentation 4](#_Toc20402263)

[3.2. Configuration Control Board 4](#_Toc20402264)

[3.3. Review and Approval Committee 4](#_Toc20402265)

[3.4. Assignment and Verification Committee 5](#_Toc20402266)

[3.5. Procedures 5](#_Toc20402267)

[**3.5.1. Change Request and Approval 5**](#_Toc20402268)

[**3.5.2. Assignment 5**](#_Toc20402269)

[**3.5.3. Verification 5**](#_Toc20402270)

[4. Software Configuration Auditing 6](#_Toc20402271)

[4.1. Baseline Verification 6](#_Toc20402272)

[4.2. Baseline Validation 6](#_Toc20402273)

# Introduction

The purpose of this document is to outline and establish the software configuration management process. The process consists of four different elements, software configuration identification, software configuration control, software configuration auditing and software configuration status accounting. This software configuration management process is tailored for the Binary Extraction and Analysis Tool (BEAT) that is currently in development.

## References

[1] Elsa Tai Ramirez, Software Requirements Specification, 2019.

# Software Configuration Identification

This section will track and identify the software configuration items that may change throughout the project lifetime.

## Software Configuration Item Identification

Here we will introduce the items that are likely to change whenever a configuration is made and types of configurations. Not all items will be changed in a given configuration; The items that do change will be mentioned for each configuration.

Items that will likely change throughout the life of the project:

1. Source code
2. Design documents
3. Test suites
4. Requirements documents
5. Project plans
6. Project standards
7. User guides

Feature configuration (i.e. Adding a new feature to the project):

1. Source code
2. Requirements documents
3. Design documents
4. User guide

Bug configuration (i.e. Fixing a bug from the project):

1. Source code
2. Test suites

Requirement configuration (i.e. A requirement has changed):

1. Requirements documents
2. Project plans
3. Project standards
4. Design documents

## Software Configuration Item Organization

This section describes the procedures that the team will use in order to protect and correctly update system files. All documents and files will be stored in GitHub, but the team will use Microsoft Teams when collaborating on documents individually.

### Labeling Schema

Naming conventions for the project will be determined by the contents of the files. On top of GitHub’s version control and file history features, we will also keep a log of versions by using GitHub commit history to determine what each version contained.

Version numbers will be documented within the files themselves until the final executables which will have version numbers displayed. New features will be incremented by one decimal. (1.10,1.20, etc.) Alterations to existing code or documents will be incremented by two decimals. (1.11,1.12, etc.) Final executables will be incremented by one full digit. (1.0,2.0, etc.). All team members have access to the main master branch, but when working on different sections, team members may create new branches with labels also correctly depicting what they are working on within that branch. All branches will also be associated with a new GitHub Issue until it is resolved and added to the master branch.

### Directory Structure

Below is the directory structure that we will be using to organize our files. Four main folders are depicted below along with a description of the contents of each folder. The team will keep folders up to date and approve all updates together.

* + - BuildControl folder - will contain approved final executable files.
    - Documentation folder - will contain updated documents relating to the project.
    - SRC folder - will contain the project code including GUI and its functionalities.
    - Testing folder - will contain project test files.

### Back-Up Plan

In order to ensure that all project files are recoverable the System Analyst, Andrea Labrado, will pull and save all working files from GitHub once every Friday into her local computer in a control folder. No alterations will be made to this file until the next scheduled pull. Andrea may also pull files more than once if the team deems it necessary to protect important files. No pulls will be made until the team ensures that all files are working and in proper order.

### Organizational Process

Each member is responsible of confirming major changes or updates with all other team members. Documents and files will be updated on GitHub following team approval.

# Software Configuration Item Organization

This section details the process and mechanisms established in order have efficient control of the software configuration. The processes and mechanisms are in place to manage who can access the project information as well as what they have access to and when it is appropriate to do so. GitHub will be the main tool to assist with the software configuration control along with additional procedures further outline in this section. In addition to this, the software configuration control aims to discern what changes are beneficial before implementation.

## Documentation

In order to maintain accountability and assess the feasibility of each change, a change request form will be submitted. The form will track several key aspects pertaining the requested changed and will require approval form a control committee. The procedures and committee outline will be presented in sections 3.2 and 3.3. Figure 1 below represents the change request form.

A screenshot of a cell phone

Description automatically generated

**Figure 1. Change Request Form**

## Configuration Control Board

This section describes the organization of the control board and the individuals assigned to specific roles. This will facilitate the approval and disapproval process for changes as well as reporting of error. Those processes will be further outlined in section 3.3

## Review and Approval Committee

The entire team constitute the review and approval committee which are responsible for reviewing any change requests and evaluating them based on appropriate criteria further outlined in section 3.3. The committee is responsible for approving or disapproving and notifying the assignment and audit committee which is listed in section 3.2.2.

## Assignment and Verification Committee

The committee consists of Alain Sanchez (Lead Programmer) and Juan Gaucin (V&V). Once a change is approved the review committee will notify the assignment and audit committee. The lead programmer will be responsible for setting up the environment for the change to be made in a process described further in section 3.3. Once a change has been requested the V&V will evaluate the change for errors and take appropriate action, this process will also be detailed in section 3.3.

## Procedures

This section outlines in detail the process and procedures for controlling changes in the software system. It is a description of the policies that be used to change the configuration of the software system.

### Change Request and Approval

1. A change request form (Figure 1, Section 3.1) will be completed and will be submitted to the review and approval committee.
2. The requestor will fill out the form to specify the following:

a. Change Name

b. Change Description

c. Type of Change

d. Priority Level

e. Date Submitted

f. Requestor Name

g. Estimated Completion Date

1. The review will be evaluated on the following criteria

a. Type of Change

b. Priority Level

c. Cost (time)

1. Approved changes will be signed off by the committee and the form forwarded to the Assignment and Verification committee for further processing.

### Assignment

1. An approved request form is received by the Assignment and Verification committee form the Review and Approval committee.
2. The lead programmer within the committee will then create an issue on GitHub for the approved change.
3. The issue will be assigned to the change requestor.
4. A type will be assigned to the issue created.
5. The request form will be attached to the issue.
6. The requestor will then generate a branch from the latest stable build to work off.
7. All work will be performed under the branch.
8. Before the change is finalized it will go to the V&V for verification to ensure there are no errors or bugs in the change.

### Verification

1. Once a change has been completed the requestor of the change will commit and push their work into the GitHub repository and notify the V&V.
2. The V&V will then verify the change and identify any error or bugs.
3. If any errors or bugs are detected, the V&V will then complete the Change Request Approval procedure to generate a change, with any suggestion for fixes if known.
4. The Assignment procedure will follow once the change is approved and assigned to the appropriate team member who submitted the original change request.

Once any change has been verified the V&V will close the issue for that change request and will be responsible merging the changes with the existing system.

# Software Configuration Auditing

This section will describe the procedures used to validate and verify that the current software system meets the documented system requirements.

## Baseline Verification

In order to verify that each software construction item is successfully implemented, each team member will first be required to submit a change request form. This form will include the issue that is started or being updated. An expected completion date will also be required. Once the form has been submitted, the auditing committee with either reject or accept it. If the form is rejected the team will discuss the reason why it was rejected and either dismiss the issue or agree on a better implementation. The individual who submitted the form is responsible of completing it within reasonable time. Once the issue has been completed the committee will verify that this new addition is correct and free of bugs before closing the issue. GitHub will keep history of the issues initiated or completed for future reference.

## Baseline Validation

Once a new feature or update has been added the team will review the software together as a whole and confirm that it is building the right system for the clients. The individuals who initiated the issue will also be required to provide a reference back to the main specification requirements. The Auditing committee will again review the changes and correctly log that these issues were resolved by some specification. In order to validate that the system is what the clients are looking for, the team will also schedule demos with the clients every two weeks to confirm that the system satisfies their requirements. Feedback will be documented and will need to be assigned to a specific member to be resolved.