Configuration Management Plan

Version Number: 1.0

Version Date: 27-Apr-2019

VERSION HISTORY

[Provide information on how the development and distribution of the Configuration Management Plan will be controlled and tracked. Use the table below to provide the version number, the author implementing the version, the date of the version, the name of the person approving the version, the date that particular version was approved, and a brief description of the reason for creating the revised version.]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version Number** | **Implemented**  **By** | **Revision**  **Date** | **Approval**  **Date** | **Description of Change** | **Reviewed By** |
| 1.0 | Aya | *01-May-2019* |  |  |  |
| 1.1 |  |  |  |  |  |
| 1.1 |  |  |  |  |  |

1. Introduction :

A [systems engineering](https://en.wikipedia.org/wiki/Systems_engineering) process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational information throughout its life.

1. Objective:

* Focuses on establishing and maintaining consistency of a product's performance, and its functional and physical attributes with its requirements, design, and operational information throughout its life.
* CM streamlines the delivery of software and applications by automating the build out of systems quickly and efficiently
* It ensures an audit trail of changes done to the system.

1. Used CM tool:

* git
* Remote repository: github

1. Configuration items:

* Source code
* Test cases
* Test plan

1. Configuration folder structure:

* Master branch: Contains the latest baseline of the project
* Development branch: contains source code , database diagram
* Testing branch: contains test cases, test plan.

1. Roles:

Configuration controller:

Implement a controlled change process.  This is typically achieved by setting up a change control board whose primary function is to approve or reject change requests that are set against any baseline.

* Testers: access “testing branch” to follow test plan, implement and execute test cases.
* Developers: access “development branch” to implement and execute the main functions of the project.
* project manager: review and accept any merge or pull request from different branches to master branch