Configuration Management Plan Version 1.0

Team01 - DEngima

October, 2017

Contents

1 Introduction

In Software engineering, software configuration management is the task of tracking and controlling changes in the software.

1.1 Purpose

Online pharmacy is a large scale project in terms of size. It is very important to identify procedures to manage these work products and the final product that will be made. A Configuration management plan document provides these procedures. These procedures help in identifying the major change and once identified, they guide us to implement and document them properly. This document describes these procedures, how to and when to perform them and responsible

members for the same. It ensures that the development is done systematically and in a disciplined manner.

1.2 Scope

The scope of this document is to identify and describe the overall policies and methods for the Configuration Management activities to be used during development life cycle, and will be updated progressively as the work proceeds and the necessity arises. The primary intention of this document is to provide overall information on the Configuration Management policy and methods to be adopted and implemented.

1.3 Intended Audience

The intended audience for this document are developing team, testers, configuration controller, team leader, etc..

2 Software Configuration Management Plan

2.1 Roles and Responsibilities

Front End Team: Whole UX and User Interface is managed by this team. If any queries or changes other members of the team are meant to contact the leader of the front end team. The sole responsibility of this team is to make sure that the user finds the interface more beautiful.

Back End Team: The whole working of the application depends on this team's work. connecting the UI to the database and making sure all the features of our application work is the sole responsibility of this team. While testing, if any feature fails to work testing team should approach the leader of this team and verify that particular feature again.

Documentation: This team makes sure that all the work done is well documented and recorded. Every document is updated in github and all the members of the group can access them through a common repository. After getting reviewed by other team members if there are any changes to be done, then the leader of this team checks whether the change is required or not. If required, the change is done and updated in github.

Project Manager: A project manager is a person who has the overall responsibility for the successful initiation, planning, design, execution, monitoring, controlling and closure of a project. Construction, petrochemical, architecture, information technology and many different industries that produce products and services use this job title.

Configuration Manager: He/She should make sure that all the changes done

in every phase (Coding, Documentation, etc..) are to be recorded. The changes done is quite a important task in developing the application, So the role of configuration manager is quite important.

2.2 Policies Applicable, Directives and Procedures

- A common repository on Github is maintained with the entire team having view and edit access to codes. They can branch from the master and work on that and can later integrate with the master so that all the updates are available immediately to the team.
- Version Control happens simultaneously when working on Github.
- All documentations are shared on a Cloud database. A primary folder where only final documents are updated is kept separate from draft versions to avoid any confusion.
- The development team must strictly adhere to the coding standards and conventions.
- All the changes are approved by the specific team leaders. for example, if there is a change to be done in the front end department the one who is requesting the change should pull a request to the team leader, whole team checks for the change, the leader decides whether the change is necessary or not.
- If the team leader thinks that a change is required, then he/she can discuss it with the team members and then decide whether to implement it or not.

2.3 Identification of Configuration Items

Configuration identification activities identifies names, and describes the documented physical and functional characteristics of the code, specifications, design, and data elements to be controlled for the project. It helps in tracking of whole processes, the work product goes through. The following section describes the details how these configuration items are identified and tracked.

2.3.1 Document Configuration Identification

Any document created during the development will act as configuration identification item and will follow a specific naming convention in a way that includes the version detail and the stage at which the document is made.
The documents will be named as:

$$documentname v_0 \ - First draft of any document.$$

$$documentname v_1.0 \ - First verified and reviewed document.$$

$$document name v_1.* - Document which has been changed but not reviewed\ verified\ after\ version 1.0.$$

$$document name v_2\ - Verified and reviewed for the second time.$$

$$document name v_n\ - Verified and reviewed for the nth time.$$

2.3.2 Software and Data Configuration Identification

- Every updated document will be uploaded on the shared github.
- Every change in the documents and code can be seen through Github.

2.4 Configuration Control

Configuration control is an important function of the configuration management discipline. Its purpose is to ensure that all changes to a complex system are performed with the knowledge and consent of management. Changes which are made without this will cause ineffectiveness which sometimes results in project failure. Configuration control tasks include initiating, preparing, analysing, evaluating and authorising proposals for change to a system (often referred to as "the configuration"). Configuration control has four main processes:

- Identification and documentation of the need for a change in a change request.
- Analysis and evaluation of a change request and production of a change proposal.
- Approval or disapproval of a change proposal.
- Verification, implementation and release of a change..

The above steps are clearly explained below.
There are three stages for any change request:
Request for Change: After a change request arrives, first and foremost update a change request document accompanying the work product so that others know that the document is soon to be changed. The, once the changes are done, update the change log and remove the request attachment. Upload the changed version.
Evaluation: Changes made to a configuration object are reviewed by assigned members. After which, the code or document is either approved, suggested some edits or additions or rejected. After the changes are done, the work product is sent for review again.
Implementation: The accepted work product is now final and is ready to be merged with the main branch of the repository or the Cloud's Final folder.

2.5 Configuration Status Accounting

The recording and reporting of information needed for configuration management including the status of configuration items (CIs), proposed changes and the implementation status of approved changes. Status accounting provides the means by which the current state of the development can be judged and the history of the development life cycle can be traced.

2.6 Configuration Auditing

A configuration management process that confirms the integrity of a systems product prior to delivery. There are two types of configuration audits:

- Functional Audit: The objective of the functional audit is to provide an independent evaluation of a software product, verifying that its configuration items' actual functionality and performance is consistent with the relevant requirement specification. This audit is held prior to software delivery to verify that all requirements specified in the Software Requirements Specification have been met.
- Physical Audit: The objective of the physical audit is to provide an independent evaluation of a software product's configuration items to confirm that all components in the as-built version map to their specifications. Specifically, this audit is held to verify that the software and its documentation are internally consistent.

Why audit the Configuration?

- All required configuration items have been produced
- All configuration items produced comply with the specified requirements
- Technical documentation completely and accurately describes the configuration items
- All approved change requests have been resolved.
- At the completion of development, the software or systems product is ready for delivery

2.7 Build Management

Build Management is the process of assembling all the components of a software application into an installable software product. This process usually includes the following steps:

- Preparing the build environment.
- Gathering of the source code.

- Labelling of the source code.
- Compilation or interpreting of the code.
- Creation of build logs.
- Creation of the installation packages.
- Updating the build statuses.

2.8 Process Management

Process management is the ensemble of activities of planning and monitoring the performance of a business process. Process management is the application of knowledge, skills, tools, techniques and systems to define, visualize, measure, control, report and improve processes with the goal to meet customer requirements profitably. In project management, process management is the use of a repeatable process to improve the outcome of the project.

2.9 Environment Management

Managing the hardware and software that hosts the system. Make sure to check whether the environment in which the software is deployed is accurate or not.Also check the server in which the application is being deployed. Prefer those servers where the crash rate of the server is pretty low. Once the software is deployed, working of thee hardware also plays an important role, so environment management also plays a crucial role in managing our application.

2.10 Teamwork

Facilitate team interactions related to the process. Make sure that each and every member of the team has an idea of the processes going on. Regular team meetings and interactions make sure that every member of the team knows his/her responsibility in making the project a successful one. If there is no proper interactions between the team members problems caused are too many. In order to reduce these problems and completing the work alloted in a short period of time team interactions are necessary.