Software Configuration Management Plan

for

Online Smart Parking system (website)

Prepared by

Kotha Brinda Vivek (19BDS0070)
Yarlagadda Pranay (19BCE0766)
Kandala Sai Krishna (19BCE0832)
Kandra Ksheeraj (19BCE0829)

Vellore Institute of Technology

10-May-2020

1. Configuration Identification and Baseline definition

(on Online Smart Parking System)

Software Configuration Management(SCM) is a process to systematically manage, organize, and control the changes in the documents, codes, and other entities during the Software Development Life Cycle. The primary goal is to increase productivity with minimal mistakes.

Tasks In SCM Process:

- Configuration Identification
- Baselines
- Change Control
- Configuration Status Accounting
- Configuration Audits and Reviews

1) Configuration Identification:

Configuration identification is a method of determining the scope of the software system. With the help of this step, you can manage or control something even if you don't know what it is.
It is a description that contains the CSCI type (Computer Software Configuration Item), a project identifier and version information.

Activities During This Process:

1)Identification of configuration Items like source code modules, test case, and requirements specification.

We have clearly distinguished the source code modules after the completion of the website.We are ready to start the testing (by using test cases).

All the source code modules along with the current version are listed below:

Instead of naming a File login.php its should be named login_v1.2.php where v1.2 stands for the version number of the file

add_category_v1.1.php
add_vehicle_v1.1.php
admin_login_v1.2.php
admin_profile_v1.1.php
bwdates_report_ds_v1.1.php
bwdates_reports_details_v1.1.php
change_password_v1.2.php
dashboard_v1.3.php
edit_category_v1.2.php
forgot_password_v1.1.php
index_v1.1.html
logout_v1.1.php
manage_category_v1.2.php
manage_incomingvehicle_v1.1.php
manage_outgoingvehicle_v1.1.php
print_v1.1.php
reset_password_v1.1.php
search_vehicle_v1.1.php
view_incomingvehicle_detail_v1.1.php
view_outgoingvehicle_detail_v1.1.php

All the folders created in the project: Instead of naming folder "vpms" it should be named "vpms_D" where D represents all the changes in the code should bebacked up daily.

vpms_D
assets_D
css_D
images_D
includes_D
js_D
admin_D
user_D

Requirement specifications should be compared to the clients requirements and also expected output functionalities.So that the client is contempt about the output of the project.We should verify that all the requirements mentioned in the SRS document are met at the end.

2)List of resources required such as the document, the file, tools, etc.

Documents/Files Required :

Software process model and Software project manage
Software Requirements Specification(SES)
Software Design Specification(SDS)
Prototyping with GUI(UML)

Tools Required In This Project:

1)Project management activity -- Lucidchart
2)Architecture diagrams – Dia
3)Detailed design – Star UML
4)Configuration management – SVN
5)Web Testing -- Selenium

2) BASELINE:

A baseline is a formally accepted version of a software configuration item. It is designated and fixed at a specific time while conducting the SCM process. It can only be changed through formal change control procedures.

A baseline is a formally accepted version of a software configuration item.

ACTIVITIES DURING THIS PROCESS:

1)Facilitate construction of various versions of an application - What features can be added in the upcoming versions of the project

In the future,we can develop improved version of our website by adding the features like
Number plate detection system by using image processing
Face identification
We can also create or develop an app by using the idea of this current version of the website.

2)Defining and determining mechanisms for managing various versions of these work products

A version control system allows users to keep track of the changes in software development projects, and enable them to collaborate on those projects. Using it, the developers can work together on code and separate their tasks through branches.
The main advantages of using a version control system include streamlining the development process, management of code for multiple projects and keeping a history of all changes within a code.

We have used one of the best version control systems GitHub to manage all the different versions of our project from the start until the end.

GitHub helps software teams to collaborate and maintain the entire history of code changes. You can track changes in code, turn back the clock to undo errors and share your efforts with other team members.

It is a repository to host Git projects. For those wondering what is Git? It is an open source version control system that features local branching, multiple workflows, and convenient staging areas. Git version control is an easy to learn option and offers faster operation speed.

2. Usage of Software Configuration Management(SCT) Tools.

➔ Configuration management tools enable changes and deployments to be faster, repeatable, scalable, predictable, and able to maintain the desired state, which brings controlled assets into an expected state.

➔ Configuration management tools make life easier for everyone using digital assets. Their primary purpose is to keep an accurate, detailed record of computer system information and standards and update them as necessary.

➔ helps us to maintain Adherence to coding conventions that make it easier to navigate code

➔ Using SCM helps in Idempotency, which means that the end state remains the same, no matter how many times the code is executed,also in Distribution design to improve managing large numbers of remote servers

➔ Some configuration management tools use a pull model, in which an agent installed on the servers runs periodically to pull the latest definitions from a central repository and apply them to the server. Other tools use a push model, where a central server triggers updates to managed servers.

➔ Configuration management tools make changes and deployments faster, remove the potential for human error, while making system management predictable and scalable, also help you to keep track of the state of your resources, and keep you from repeating tasks, like installing the same package twice.

➔ Improve system recovery after a critical event with automated configuration management. If a server goes down for an unknown reason, you can deploy a new one quickly and have a record of any changes or updates that occurred so you can identify the source of the problem.

➔ configuration management tools can also help you to run an audit of your system so you can more quickly identify where the problem is coming from.

➔ SCM tools look after:

- Configuration identification - Identifying configurations, configuration items and baselines.

- Configuration control - Implementing a controlled change process. This is usually achieved by setting up a change control board whose primary function is to approve or reject all change requests that are sent against any baseline.
- Configuration status accounting - Recording and reporting all the necessary information on the status of the development process.
- Configuration auditing - Ensuring that configurations contain all their intended parts and are sound with respect to their specifying documents, including requirements, architectural specifications and user manuals.
- Build management - Managing the process and tools used for builds.
- Process management - Ensuring adherence to the organization's development process.
- Environment management - Managing the software and hardware that host the system.
- Teamwork - Facilitate team interactions related to the process.
- Defect tracking - Making sure every defect has traceability back to the source.

➔ Some benefits of using SCM tools are:

- Diminished likelihood of outages and security breaches
- Cost-effective due to avoiding technology asset duplication
- Tighter process control by enforcing established policies and procedures and Quicker problem resolution
- Efficient change management by reliance on the baseline configuration and Faster service restoration turnaround

➔ Configuration management tools in DevOps handle crucial tasks such as deploying applications, maintaining infrastructure, and provisioning environments. These functions are delicate and labor-intensive, but the tools can save time and reduce the chance of human error if they are automated.

3. Some Magnificent Configuration Management(SCM) tools

1) Desktop central

Desktop Central offers configurations that help administrators manage applications, system settings, desktop settings, and security policies. It can be used to deploy a group of configurations all at once using the collection feature.

Features:

- It offers over 100 ready-made custom script templates and 30 pre-defined configurations, ready to be deployed anytime
- It also provides computer-based and user-based configurations.
- Offers configurations for Mac and Linux as well.
- Status of the applied configurations can be tracked anytime.

2) CFEngine Configuration Tool

CFEngine is a configuration management framework. It allows you to manage your mission-critical tasks securely. This system configuration management tool is available as both open source and commercial software.

Features:

- This is the best configuration management tool that Helps you to define the desired state and configuration of your IT infrastructure
- Allows you to automate roll out updates and changes to every node in your infrastructure
- It uses autonomous agents which run on every node of your infrastructure.

3) Server Configuration Monitor

Server Configuration Monitor is a tool to detect and compare configuration changes to the server, applications, and databases. It allows you to configure applications and servers in real time.

Features:

- The tool provides alerts and reports on deviations.
- You can compare the current configuration with previous versions.
- It allows you to track server software and hardware inventory.
- This app can be used to monitor output changes.
- It can be integrated with Orion platform (bandwidth performance and fault management tool)

4) Puppet Configuration Tool

Puppet is an open source and Best configuration management tool for centralizing and automating the configuration management process. It is used to configure, manage, deploy various applications and services.

Features:

- Automate provisioning across your IT infrastructure
- This scm tool Allows you to take full control and visibility over your software delivery process.
- Allows you to make quick changes or remediate urgent problems alongside your model-driven automation management.
- This best configuration management tool Helps you to manage infrastructure as code using your favorite version control systems

5) CHEF Configuration Tool

The chef is an automation platform that offers a method to configure and manage infrastructure. In this tool, Infrastructure as code implies by executing coding instead of performing manual execution. The chef tool works on Ruby and DSL for writing the configurations.

Features:

- Chef follows the Push model and offers an easy cloud adoption.
- This is the best configuration Management Software helps you to increase service resiliency, to develop more defect-free software.
- Chef offers automation abilities which helps you to reduce the risk and improve compliance at all stages of development.
- Configure various cloud-based SaaS services, and integrate cloud provisioning APIs and third-party software.