

Ho Chi Minh Tour Guide

CM Plan

Project Code: HTTPS

Document Code: HTTPS\_CM – v0.1

Note: Text displayed in blue italics is included to provide guidance to the author and should be deleted or hidden before publishing the document.

This template can be used at it is, or to complete and improve an already existing template.

Help:

* Don’t remove headlines level 1 and level 2 (Heading1 and Heading2)
* Reasons why a section is not applicable shall be documented under the respective headline

**Ho Chi Minh, 11/1/2013**

SIGNATURE PAGE

Author:

HieuHT, Title Date

TuanBT, Title Date

TuanNH, Title Date

PhongTP, Title Date

SinhCV, Title Date

REVIEWERS:

PhuongLHK, Instructor Date

APPROVAL:

PhuongLHK, Instructor Date

Record of change

\*A - Added M - Modified D – Deleted

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Effective Date | Changed Item | A\* M, D | Change Description | Reason for Change | Revision Number |
| 11/1/13 | HTTPS\_CM – v0.1 | A | Create new document |  | V0.1 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

TABLE OF CONTENTS

[1. Introduction 3](#_Toc213570602)

[1.1. Role & Responsibility 3](#_Toc213570603)

[1.2. Definitions and Acronyms 3](#_Toc213570604)

[2. configuration management Process 3](#_Toc213570605)

[2.1. CI definition, Naming convention & Storage 3](#_Toc213570606)

[2.2. Movement of CIs through their storage 3](#_Toc213570607)

[2.3. Directory structure & Access right 3](#_Toc213570608)

[2.4. Version numbering rule 3](#_Toc213570609)

[2.5. Change control 3](#_Toc213570610)

[2.6. Baseline plan 3](#_Toc213570611)

[2.7. Backup strategy 3](#_Toc213570612)

# Introduction

The purpose of this document is to identify and describe configuration management (CM) process implementing in the project HTTPS\_CM – v0.1

## Role & Responsibility

Refer to Project Organization section in Project Plan

## Definitions and Acronyms

| Acronym | Definition | Note |
| --- | --- | --- |
| ADD | Architecture Design Document |  |
| CC | Infrastructure Configuration Controller |  |
| CI | Configuration Item |  |
| CM | Configuration Management |  |
| CSCI | Computer Software Configuration Items |  |
| DDD | Detail Design Document |  |
| PM | Project Manager |  |
| PTL | Project Technical Leader |  |
| PIC | Person in Charge |  |
| QA | Quality Assurance Officer |  |
| SRS | Software Requirement Specification |  |
| Source | Source Code |  |
| URD | User Requirement Document |  |
| TP | Test Plan |  |
| TC | Test Case |  |
| WIP | Work in Progress |  |
| WP | Work product |  |

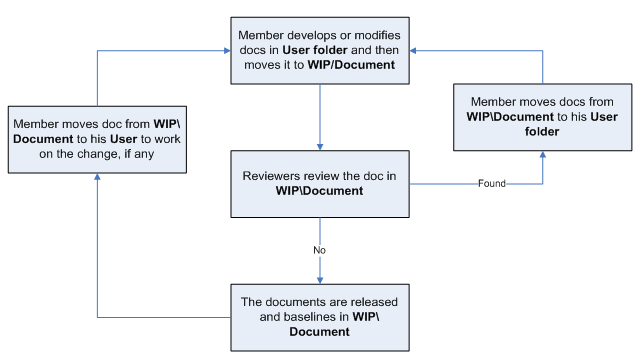
# configuration management Process

## CI Identification & Naming convention

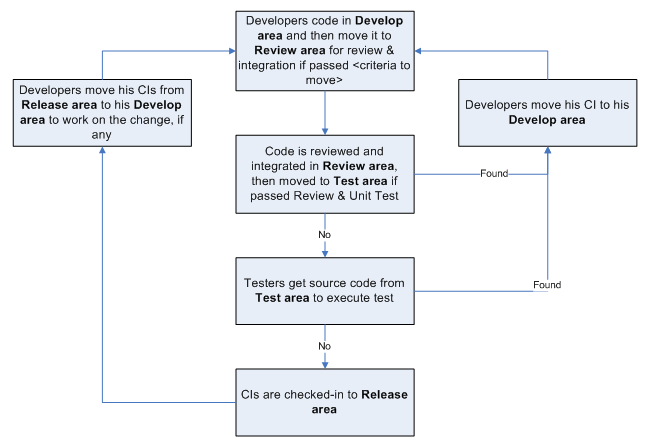
Refer to sheet CI Identification in CM Report

## Movement of CIs through their storage

#### For Document



#### For Source code:



## Directory structure & Access right

### Promotion Areas

| Area | Purpose |
| --- | --- |
|
| Develop Area | Area for different users to store his/her owned items |
| Review Area | To store items that is ready for review.  Reviewer get to be-reviewed items from this area |
| Test Area | Just applicable for Source items.  To store items passed Unit Test and Code Review |
| Release Area | To store the items ready for release and all released versions of items  Users get the most recent items for their usage from this area |
| Archive Area | To archive all released versions of each CI |

### Directory structure

| Main Folder | Sub Folder | Purpose | Map to Area | Access right |
| --- | --- | --- | --- | --- |
| **Project Directory :** https://https-project.googlecode.com/svn/trunk/ | | | | |
| WIP | Deliverables | Store all CIs that are delivered to customer, be possible to add date to folder name | Release | Full: PM, CC  Modify: None  Read: All |
| Documents | Documents of Requiements, Design, Test, … | Release + Review | Full: PM, CC  Modify: PIC  Read: All |
| Meeting minutes | Store project meeting minutes, including meeting minutes with customer | NA | Full: PM, CC  Modify: All |
| Plan | Store Proposal, Estimation, Project Plans, Project schedule, Task list | Review + Release | Full: PM, CC  Modify: PTL  Read: All |
| Report | Store Project Reports: Weekly , Milestone, Post-mortem, Acceptance note, other Event-driven reports | NA | Full: PM, CC  Modify: PTL  Read: All |
| Record | Store project records, divided into  Review: include Review, Test and Inspection records  Change request  Acceptance  Mails  ... | NA | Full: PM, CC  Modify: All |
| Source | Store VSS file of Source code | Archive | Refer to VSS directory |
| User | User’s working area, store user’s owned items | Develop | Full: PM, CC  Modify: User  Read: All |
| Reference | Customer supplied | store Documents and Other materials/data supplied by customer or those support software development and production operation in the project… | Release | Full: PM, CC  Modify: PIC  Read: All |
| Process Template | Store Guidelines/Standards/Forms/Templates/Checklist specified for the project usage | Release |  |
| Audit |  | Store QA work products  Process review  Final inspection  Work product review | NA | Full right: Project QAs  Read right: All |
| Archive | Baseline Name | To released versions of CIs at baselines | Archive | Full: PM, CC  Read: All |
| **VSS Directory:** N/A | | | | |

#### For Physically stored items

|  |  |  |  |
| --- | --- | --- | --- |
| Items | Location | Person in charge | Usage rule |
| N/A |  |  |  |

## Version numbering rule

#### For Documents:

The version level is maintained as numbered identifier with two components

1.1

Revision

Version

The original version will be numbered 0.1. Subsequent revisions will be numbered 0.2, 0.3. The release version will be 1.0.

**Version number**, which appears to the left of the decimal. It changes only when the core architecture of the item changes. For example: when an item is completely overhauled, with substantial internal changes, the version 1.0 would become version 2.0

**Revision number**, which appears to the right of the decimal. It changes when existing content is changed, but the overall structure and flow of the item remains the same. The normal sequence of revision is 1.1, 1.2 and so on.

#### For Software Source Files:

Software executables and support files are generally identified by name and version number, such as “Main DB v1.1.a”. The scratch edition will be 1.0. The version numbering scheme consists of three components:

1.1a

Revision

Version

Update

view

**Version number**, which appears to the left of the decimal. It changes only when the core architecture of the software item changes, as when moving from one area of the development tool to another, when an application is completely overhauled, or the user interface changes fundamentally. In this case, version 1.1a would become version 2.0.

**Revision number**, which appears to the right of the decimal. It changes when new features, functionality or other content are added or significantly changed. In normal case, the core architecture or user interface have been extended or limited in some manner. The most common reason for changing the revision number is when adding a new module or other functionality to the software. The normal sequence of revision is 1.0, 1.1, and 1.2 and so on.

**Update level** is appended or incremented when the only change to the software item is to correct one or more defects, without the addition of any new functionality. Version 1.1 would become v1.1a, v1.1b and so on. This updating is over ridden when a combination revision, involving bug fixes and new feature additions, is performed. In such a case, the software revision number is incremented and any update indicator is dropped, as in v1.1b to v1.2