|  |
| --- |
| Configuration Management Plan |
| Human Resource Management |
| Define source files revisions, the repository where all revisions of all files necessary to create any product version can be found |

6/6/2012

Table of Contents

[1. Introduction 1](#_Toc326743548)

[1.1 Purpose 1](#_Toc326743549)

[1.2 References 1](#_Toc326743550)

[2. Software Configuration Management 1](#_Toc326743551)

[2.1 Organization, Responsibilities, and Interfaces 1](#_Toc326743552)

[2.2 Tools, Environment, and Infrastructure 1](#_Toc326743553)

[3. The Configuration Management Program 1](#_Toc326743554)

[3.1 Configuration Identification 1](#_Toc326743555)

[3.1.1 Identification Methods 1](#_Toc326743556)

[3.1.2 Project Baselines 1](#_Toc326743557)

[See more in project plan 1](#_Toc326743558)

[3.2 Configuration and Change Control 1](#_Toc326743559)

[3.2.1 Change Request Processing and Approval 1](#_Toc326743560)

[3.2.2 Change Control Board (CCB) 2](#_Toc326743561)

[3.3 Configuration Status Accounting 3](#_Toc326743562)

[3.3.1 Project Media Storage and Release Process 3](#_Toc326743563)

[3.3.2 Reports and Audits 4](#_Toc326743564)

**Revision Table**

|  |  |  |  |
| --- | --- | --- | --- |
| **Author** | **Date** | **Reason for changes** | **Version** |
| Tan Tran | 11/11/2011 | Add Configuration Management Plan | 1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The purpose of Software Configuration Management:

* + Create the repository where all revisions of all files necessary to create any product version can be found
  + Easy to define which source files revisions were used in each build so developers don’t waste a lot of time tracking down bugs in other people’s code that prevent them from testing
  + Easy to know what changed in each revision of our source files, and why
  + Can recreate previous product releases
  + The QA team has a stable product to work with

## References

The project plan

# Software Configuration Management

## Organization, Responsibilities, and Interfaces

See more in project plan

## Tools, Environment, and Infrastructure

Google code and SVN for creating the repository of all documents in project

# The Configuration Management Program

## Configuration Identification

### Identification Methods

* The configuration item’s name format: [HRM] Document name
* In each configuration item, team members need to log the version in revision table at the beginning in each document.

The version format: XX.YY.ZZZZ

XX: is the Major Release Number

YY: is the Minor (Maintenance) Release Number

ZZZZ: the build number

* + For every change from team members in document or configuration items, the ZZZZ (build number) will be increased.
  + After each baseline (after review with mentor or customer), the YY will be increased
  + Whenever the changes in configuration item are too large, the XX will be increased

### Project Baselines

### See more in project plan

## Configuration and Change Control

### Change Request Processing and Approval

CHANGE REQUEST MANAGEMENT

Type | Classification | Template

VALIDATION

DEVELOPMENT

PROJECT ANALYSIS

Validate | Close

Requirements | Architecture | Design | Code | Test

Planning | Approval | Impact Analyses | Prioritization

**MANAGEMENT**

Assets & Activities | Versioning | Baseline | Security

*Figure 1: Change request process*

**PROCESS DESCRIPTION**

|  |  |
| --- | --- |
| **Phase name** | **Description** |
| **Project Analysis** | When the changes happen, the team member will analyze the changes in many aspects like the benefit and bad effect when we embrace the change. Base on the impact and result when the changes happen, the team members will decide to approve the changes or not.  With the enhancement change, the customers/mentor must prioritize the changes and the team will start implement the highest-prioritized changes. It like a stack and there any new changes will add to stack, the changes may be re-prioritized and they can be removed at anytime |
| **Development** | When the changes are approved, the project team will start to implement from the highest-prioritized changes to the low – prioritized one. |
| **Validation** | The customer and development team will validate if the changes implement correctly. If it is correct, close the process |
| **Management** | All of phases are managed by the project manager (PM).  + Assets and Activities  + Versioning: Build configuration management team.  + Baseline: is very important to manage changes and negotiate with the customers |

.

.

### Change Control Board (CCB)

|  |  |  |
| --- | --- | --- |
| **Team member name** | **Role** | **Responsibility** |
| Nhung Huỳnh | Quality Assurance | Ensure that all changes is implement well.  Manage the meeting for change analysis |
| Lộc Phan | Writer | Log all changes in project and report at the meeting.  Write the change analysis meeting report |
| Tùng Nguyên | Test leader | Detect the change or defect and report to writer for logging. |
| Nguyen Dinh | CCB team member | Analysis and contribute idea on implement change or not |
| Tan Tran |
| Tuong Nguyen |
| Đăng Nguyen |

## Configuration Status Accounting

### Project Media Storage and Release Process

**Accepted change request**

**Check out**

Failed

**Deploy change**

**Integrate test**

**Update to new version**

**Unit test**

Passed

**Check in**

*Figure 2: Project media storage and release process*

The input of process is the accepted changes request. There are two kind of change in HRM project

* Bug changes: The developer trust to fix without ask for permission
* Enhancement changes: Team member review and mentor/customer review and produce the list of changes.

Whenever the request is accepted, the developers will check out the source file from the repository. Repository is where all source file is kept after each changes.

The developers make change in their own computer and execute the unit test.

The developer after change source code must check the new version in repository. If there is also new version of source file, the developer must check out new version and update their changes into new version. After that, they can check in new source code to repository. That is new revision.

Integration test is also implemented in each revision. If the integration test is passed, the new revision is created.

### Reports and Audits