UNIVERSITI KUALA LUMPUR

SCM Project

ISB 42303

SOFTWARE CONFIGURATION MANAGEMENT

Yellow Givers Online Donation System

Software Configuration Management Plan
Version 1.0

Revision History

Date	Version	Description	Author
30/05/201	**1.0**	**FirstVersion**	**Muhamad Yusof Bin Puteh**

Document Approval

Authorization	Name	Date
Prepared By : ________________________ Configuration Manager	MUHAMMAD SYAHMI BIN SHAFAWI MUHAMAD YUSOF BIN PUTEH	6/5/2017
Approved By : ________________________ Project Manager / Change Control Manager	MOHD FAHRUL FAHLEVI BIN FUAD MUHAMMAD SHAHZWAN BIN MOHD ZAKI	8/5/2017
Approved by: ________________________ SCM Lecture	DR. AEDAH BINTI ABDUL RAHMAN	

Software	: Yellow Givers Online Donation System – Version 1.0
Archiving Place	: -
Copies Available	: Yellow Givers Online Donation System – Version 1

Table of Contents

Figures

Configuration Management Plan

1. Introduction

The Software Configuration Management (SCM) Plan specifically addresses configuration management for software. The detail described about Configuration and Change Control Management (CCM) activities to be performed during the course of project. It details the schedule of activities, the assigned responsibilities and the required resources including staff, tools and computer facilities.

However, this document may contain information of a sensitive nature. This information should not be given to persons other than those who are involved with this system/project or who will become involved during its lifecycle.

The configuration Management of project artifacts will be addresses separately.

1.1 Purpose

The purpose of Software Configuration Management (SCM), in general, is to establish and maintain the integrity of work products using:

- Configuration Identification
- Configuration Control
- Configuration Status Accounting
- Configuration Audit

A Configuration Item (CI) is an entity designated for configuration management, which may consist of multiple related work products that form a baseline. This logical grouping provides ease of identification and controlled access. The selection of work products for configuration management should be based on criteria established during planning. Section 3 of this SCM Plan contains detailed information about CIs.

Configuration Identification

The purpose of Configuration Identification is to define the functional and physical characteristics of a CI in sufficient detail so that it may be developed, tested, evaluated, produced, competitively procured, accepted, operated, maintained, and supported. Configuration Identification is established by baselines plus approved changes. For purposes of this SCM Plan, Configuration Identification includes the selection, creation, and specification of the following:

- Products that are delivered to the client
- SEM documents requiring Structured Walkthroughs (SWT)

Configuration Control

Configuration Control is the process of evaluating, approving or disapproving, and managing changes to controlled items. This includes tracking the configuration of each of the CIs, approving a new configuration if necessary, and updating the baseline.

Configuration Status Accounting

Configuration Status Accounting is theprocess of creating and organizing the information necessary for the performance of configuration management. An element of configuration management consisting of the recording and reporting of information needed to manage a configuration effectively. This information includes a listing of the approved configuration identification, the status of proposed changes to the configuration, and the implementation status of approved changes.

Configuration Audit

Configuration Audits are conducted to verify that a CI, or a collection of CIs that make up a baseline, conforms to a specified standard or requirement. This includes functional and physical configuration audits.

1.2 Objectives

This SCM Plan defines the configuration management policies and procedures required for this project. This plan has been developed early in the lifecycle to ensure the control of changes as soon as the project requirements are approved. This plan addresses activities that are platform independent, such as identifying the items that will be placed under configuration management. As the project progresses through the lifecycle stages, the plan is expanded to reflect platform specific activities.

Changes in this system affecting other SCM plans are identified and explained in Section 2 (Software Configuration Management) and Section 3 (Software Configuration Management Program) of this plan.

1.3 Definitions, Acronyms and Abbreviations

Term	Definition
Configuration Management	Configuration management encompasses the administrative activities concerned with the creation, maintenance, controlled change and quality control of the scope of work.

Abbreviation	Definition
CM	Configuration Management
CI	Configuration Item
VCS	Version Control System

1.4 References

Listed here are policies, procedures and standards used in preparing and setting up this SCM Plan.

- IEEE Standard for Software Configuration Management Plans, IEEE Std 828-1990.
- Example Template : http://www.upedu.org/templates/cs/CCM/upedu_ex_cmpln.pdf
- SCM Course Study Material ISB42303: https://vle.unikl.edu.my/course/view.php?id=13018

1.5 Overview

At first, the document explains the software configuration management environment then goes on with the configuration management program (including Configuration Identification, Configuration and Change Control and Configuration Status and Accounting). More information is provided on the miles tones strategies, the training and resources used and the external development environment.

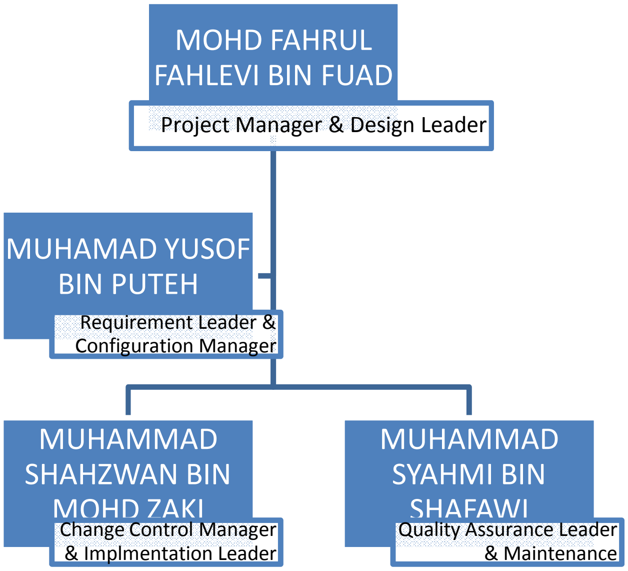
2. Software Configuration Management (SCM)

2.1 Organization, Responsibilities, and Interfaces

This section identifies the roles of individuals and groups that participate in the SCM process. It describes the relationships between individuals and groups.

2.1.1 Roles & Responsibilities

All team members participating to the CM activities. Although, Muhamad Yusof Bin Puteh has been assigned the official Configuration Management Manager and Muhammad Shahzwan Bin Mohd Zaki assigned for Change Control Manager. Details of the roles and responsibilities by each team member are as per below:


MOHD FAHRUL FAHLEVI BIN FUAD
Project Manager & Design Leader
MUHAMAD YUSOF BIN PUTEH
Requirement Leader & Configuration Manager
MUHAMMAD SHAHZWAN BIN MOHD ZAKI
Change Control Manager & Implmentation Leader
MUHAMMAD SYAHMI BIN SHAFAWI
Quality Assurance Leader & Maintenance

Project Manager & Change Control Team

<u>Project Manager</u>

- Establish the overall project schedule for SCM activities with Configuration Management Manager (CMM)
- Validates that team members have been trained in and knowledgeable of SCM concepts and techniques and that they are applied to project activities
- Ensure compliance with the SCM standards and procedures set by the CMM, the Change Control Board (CCB), and any other affected groups as outlined in this plan
- Participate as a member of the Change Control Board
- Allocates resources, shapes priorities, coordinates interactions with customers and users, and generally keeps the project team focused on the right goal.
- Establishes a set of practices or standards that ensure the integrity and quality of project artefacts.

Change Control Manager

- The change control manager role oversees the change control process. This role is usually played by a Configuration (or Change) Control Board (CCB) and consists of representatives from all interested stakeholders.
- The change control manager is also responsible for defining the Change Request Management Process, which is documented in the Change Management Plan.
- The change control manager should understand configuration management principles. He or she should be skilled in estimating cost and schedule impacts of change requests.
- Also should be able to communicate effectively in order to negotiate scope changes and in order to determine how each change request should be handled and by whom.

Requirement Management Leader

- Leads and coordinates requirements elicitation and use-case modelling by outlining the system's functionality and delimiting the system such as establishing what actors and use cases exist, and how they interact and shall ensure its satisfy stakeholder needs.
- Knowledge of the business and technology domains is essential to have amongst those acting in this role.

Design Leader

The designer role defines the responsibilities, operations, attributes, and relationships of one or several components, and determines how they will be adjusted to the implementation environment.The designer must have a solid working knowledge of:

- Use-case modeling techniques
- System requirements
- Software design techniques, including object-oriented analysis and design techniques, and the Unified Modeling Language
- Technologies with which the system will be implemented

Implementation Leader

- The implementer role is responsible for developing and testing components, in accordance with the project's adopted standards, for integration into larger subsystems.
- When test components, such as drivers or stubs, must be created to support testing
- The implementer is also responsible for developing and testing the test components and corresponding subsystems.

Quality Assurance Leader

Responsible for the core activities of the test effort, which involves conducting the necessary tests and logging the outcomes of that testing. This covers:

- Identifying the most appropriate implementation approach for a given test
- Implementing individual tests
- Setting up and executing the tests
- Logging outcomes and verifying test execution
- Analyzing and recovering from execution errors

Configuration Management Leader

- The configuration manager provides the overall Configuration Management (CM) infrastructure and environment to the product development team.
- The CM role supports the product development activity so that developers and integrators have appropriate workspaces to build and test their work, and so that all artifacts are available for inclusion in the deployment unit as required.
- The configuration manager also has to ensure that the CM environment facilitates product review, and change and defect tracking activities.
- The configuration manager is also responsible for writing the CM Plan,identification of CIs and providing configuration standard and templates to the project team

2.2 Tools, Environment, and Infrastructure

All project source code and baseline document will be stored in a Github Repository to access by the team member for better coordinating among the team members and will be available at any time at the following address.

Host: https://github.com/yellowgivers

Login (email) : yellowgivers@gmail.com

Password: abc123456

3. The Configuration Management Program

3.1 Configuration Identification

The terms Configuration Identification and Configuration Item are defined in Section 1.1 of this document.

In this SCM Plan, work products are considered for configuration management based on the following criteria. A work product is any tangible item that results from a project function, activity or task.

- May be used by one or more work groups
- Are expected to change over time either because of errors or change of requirements
- Are dependent on each other in that a change in one mandates a change in another/others
- Are critical to the project

3.1.1 Identification Methods

Each of product artifacts / baseline produce each after development stage are to be named following this rule:

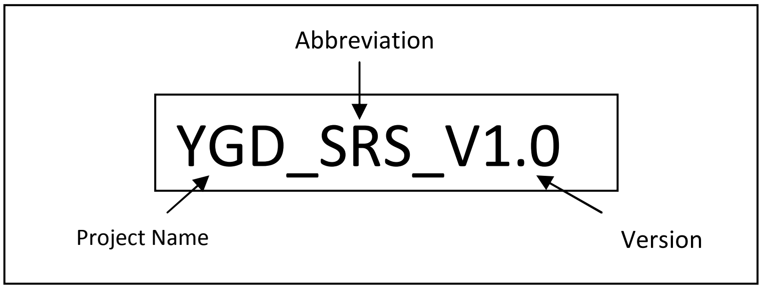


Figure1 : Artifacts' Identification Method

3.1.1.1 Artifacts Abbreviations Policies

Artifacts' abbreviation must always have 3 characters. Also, always use lower case characters for non – starting letters. Here are some examples:

Single – word artifacts: Glossary (Glo)
Two – word artifacts : Project Proposal (PrP)
Three – word artifacts : Software Requirement Specification (SRS)

3.1.1.2 Artifacts to Produce

Here is a list of all artifacts' or baselines and their abbreviation that are produce in each of development stages and will be put under configuration item

Artifacts / Baselines	Abbreviation
Glossary	**Glo**
Prototype	**Pro**
Reference	**Ref**
Change Request Report	**CRR**
Configuration Management Plan	**CMP**
Software Requirement Specification	**SRS**
Software Design Document	**SDD**
Software Testing Plan	**STP**
Operation & Maintenance Manual	**OMM**
Project Proposal / Plan	**PrP**
Test Cases	**TeC**
Test Components	**TsC**
Test Results	**TeR**
User Manual	**UsM**
Sample Template	**SaT**

3.1.2 Product Directory Structure (Project Repository)

The project directory that will be use for Yellow Givers Online Donation System will be available via Github as described under Section 2.2.This includes all the source code and baseline for each of development stages

The Project Referential is defines as in section 3.1.2.3

3.1.2.1 Workspaces

Development Workspace: All development spaces are private to each of team members and they will use their own workstation or computer a development workspace. Accesses to the integration workspace or baseline are going through GitHub a version control repository as mention in Section 2.2

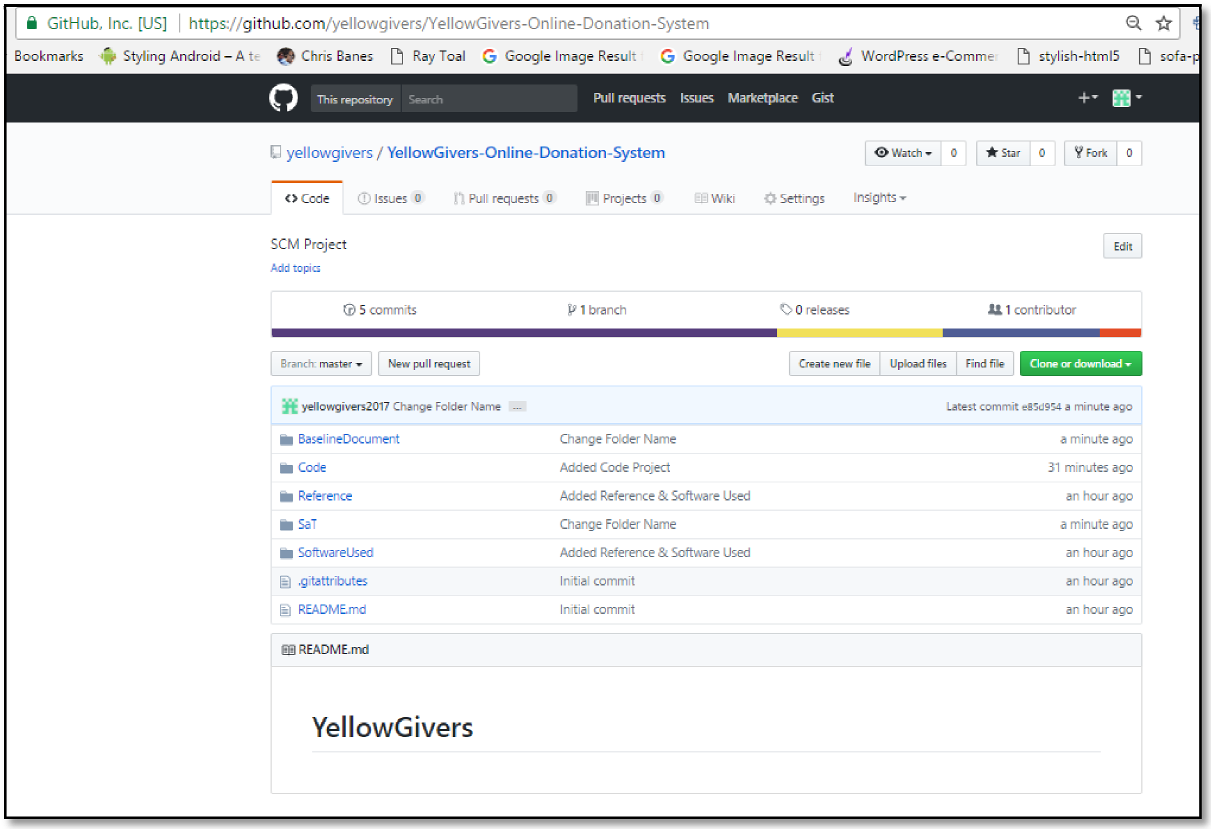
Integration Workspace: As the project repository is hosted on Github site, this site will be acting Integration Workspace too. Baseline policies are defined in Section 3.1.3.All project artifacts and baselines are accessible through the Github.This is where overall product is to be built.

3.1.2.2 Product Directory Naming Conventions

Each of the artifacts or baseline are arrange according to their folder in Integration Workspace. If the section is more than one word long, the words will be concatenated.

3.1.2.3 Baseline Referential

The Baseline Referential follows this structure as illustrate in Figure 2.



SCM Project

yellowgivers2017 Change Folder Name … Latest commit e85d954 a minute ago

BaselineDocument	Change Folder Name	a minute ago
Code	Added Code Project	31 minutes ago
Reference	Added Reference & Software Used	an hour ago
SaT	Change Folder Name	a minute ago
SoftwareUsed	Added Reference & Software Used	an hour ago
.gitattributes	Initial commit	an hour ago
README.md	Initial commit	an hour ago

README.md

YellowGivers

Figure 2: Referential Directory Structure

3.1.3 Project Baselines

Whenever a team member wants to work on the artifact he / she will clone (check out) the configuration item into his own workstation and do modification to the artifact. Once he finished modification, he or she will then commit the changes and then push to origin master back to the control library which is the Github repository.

GitHub and similar services bring all of the benefits of a decentralized VCS to a centralized service. GitHub also stores a copy of the project's repository just like any other developer. Then, team basically designate that as the project's central repository and all the developers push and pull their changes to and from that repository. GitHub takes this a step further by encouraging developers to fork a project's repository and then use that as their own centralized repository. From there the team can send "pull requests" to the main project with their changes and then the project manager can review them before deciding whether to include them in their project or not.

After each iteration, a build is produced and put the baseline in the project repository.

3.1.3.1 Special folder for current code

The testers will need the current code files to be available at any time.Thus,special folder, named “Construction”, is used for that matter.This folder is a sub-folder of the “Code” folder in the project directory structure. All coded files are contained within this folder.

3.2 Configuration and Change Control

3.2.1 Change Request Processing and Approval

The only team member authorized to add new reference documents or folder to the project referential is the configuration manager and only this member can modify the content (structure) of the referential. Muhamad Yusof Bin Puteh has been assigned to this function and applies the following steps when submitting an artifact on the referential:

- Quickly revise the Change Request document
- Put the document in the “ToBeRevise” folder (External to the referential)
- Alert and advice all team member that a Document Review are required (mostly using email or telephone). Reviews will take place mostly on Saturday between 8:30am and 2:30pm.
- Document modifications are iterative and distributed among team members.
- When the document is accepted by all reviewers, Configuration Manager will transfers the softcopy of document in its associated folder to the project referential which is the Github

3.3 Configuration Status Accounting

3.3.1 Project Media Storage and Release Process

Backup storage is made through another website, this one located in the Google Drive. The configuration manager will ensures that all Yellow Givers Online Donation System versions are stored and backed up on this Google Drive,using the following login information:

Host: https://drive.google.com/drive

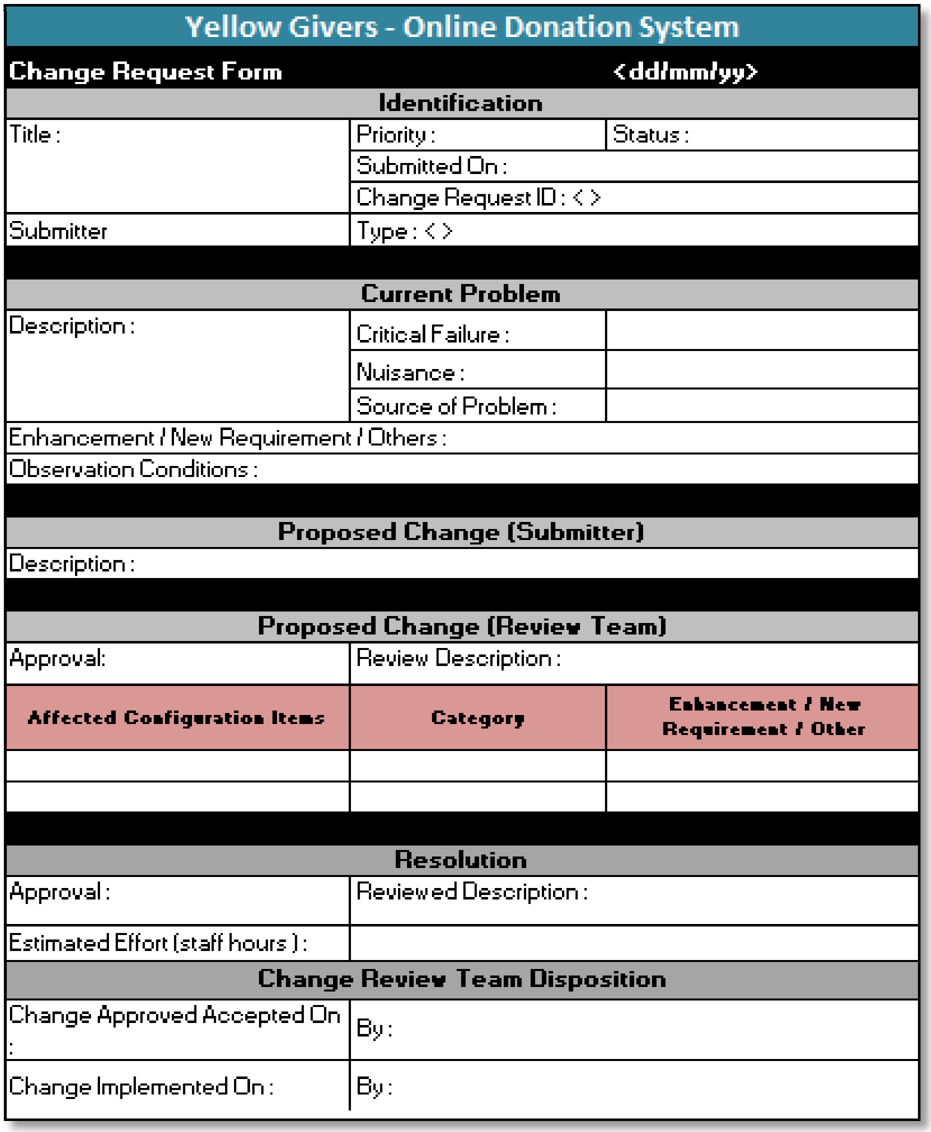
Login (email) : yellowgivers@gmail.com

Password: abc123456

Security backups are made every Friday, after the reviews between 5:30pm and 6:00pm. This backup version of the project will also be used by the project manager for follow-ups.

3.3.2 Reports and Audits

Only one report is used in the Yellow Givers Online Donation System which is the Change Request Report. Below is the change Request Form that will be in this project.



Yellow Givers - Online Donation System		
Change Request Form	**<dd/mm/yy>**	
Identification		
Title :	Priority :	Status :
	Submitted On :	
	Change Request ID : < >	
Submitter	Type : < >	
Current Problem		
Description :	Critical Failure :	
	Nuisance :	
	Source of Problem :	
Enhancement / New Requirement / Others :		
Observation Conditions :		
Proposed Change (Submitter)		
Description :		
Proposed Change (Review Team)		
Approval:	Review Description :	
Affected Configuration Items	**Category**	**Enhancement / New Requirement / Other**
Resolution		
Approval :	Reviewed Description :	
Estimated Effort (staff hours) :		
Change Review Team Disposition		
Change Approved Accepted On :	By :	
Change Implemented On :	By :	

4. Milestones

The Configuration Management Plan should be updated after each iteration of Software Development Stages.

5. Training and Resources

Software tools: Microsoft Office Suite, GitHub, Any Email Client, availability of any used templates and documentation.

GitHub Training has been organized by Software Configuration Management Class on 22 April 2017

6. External development Environment

Refer to Section 2.2.