CENG 491

DIGIMOD
Configuration Management Plan

GROUP NAME: **MANAS YAZILIM**

GROUP MEMBERS: **Gökhan ÇAPAR**

Hüseyin B. AYYILDIZ

A. Serhat DEMİR

Seltchouk AHMED

1. Introduction

1.1. Purpose of Software Configuration Management Plan

Development of software is a long and complicated process which is actually a team work. During this process things tend to change independent of how perfect it is designed. Since changes and updates are inevitable they have to be controlled by analyzing work products which are likely to change, creating relationships between components of product and determine a mechanism to handle different versions.

In such a long term, team made development process configuration management plan becomes crucial. A jargon should be developed and all changes should be handled with a known methodology, which configuration management plan will provide, because absence of these will cause catastrophic complications and errors which may endanger completion of the product in time.

The primary goal of this document is to support the development process. It is to enhance and improve product and service delivery to final users. One of the major goals is to change the mainstream development process from one of a systematic production to one of new carefully designed releases. This includes the concept of versioning for software applications that comprise the majority of systems or other products.

1.2. Scope of Document

This document is the plan of configuration management of DIGIMOD which is currently developing by Manas Yazılım.

This plan documents the Configuration Management activities that will be used by Manas Yazılım during the implementation of DIGIMOD. This plan defines what will be put under configuration management control, who will be involved in the configuration management process, what procedures are to be used, and what timeline has been planned for these activities.

1.3. Definitions, Acronyms and Abbreviations

CI Configuration Item

CM Configuration Management

CSA Configuration Status Accounting

CVS Concurrent Versioning System

SCA Software Configuration Audit

CMP Configuration Management Plan

SCMP Software Configuration Management Plan

1.4. Document Overview

In the first chapter we introduced what CM is and why it is essential for our project.

In the second chapter we declared responsibilities of members in the project, how is our organization to create our CMP.

In the third chapter CI's are identified and methodology that will be followed for CM is created.

In the fourth chapter we declared our milestones and their deadlines.

In the fifth chapter resources needed for CM are declared and explained.

In the sixth chapter optimization of CMP is explained.

1.5. Benefits of This Document

This CM plan will be beneficial for our development process by providing:

- Integrity of items is maintained throughout the project lifecycle.
- Changes are effectively controlled and managed.
- Quality is improved by more effective configuration management.

1.6. Document References

IEEE Standard (IEEE Std 828-1998) for Software Configuration Management Plan is the main reference we have followed while creating this document.

2. Software Configuration Management

2.1. Organization

In our project, we would need three teams. Configuration management team, development team and test team. Our whole crew consists of 5 people and all people will take parts in all teams. Although it looks like we are giving different names to same team, it is necessary to define our responsibilities.

2.2. Responsibilities

Configuration Management Team is responsible from:

- Preparation of configure management plan
- Management of configuration items
- Informing development team about an configuration item
- Approve or reject a change request
- Checking correctness of a change after it is made
- Determine baselines
- Perform audits

Development Team is responsible from:

- Implementing configuration items
- Implementation of changes

Test Team is responsible from:

- Testing development teams work
- Request changes when necessary

2.3. Tools and Infrastructure

We will use concurrent versioning system (CVS) for our commonly used files and a common ftp area for other files.

3. Configuration Management Process

3.1. Identification

- Software: Software configuration item is basically our developing program which consists of 5 modules.
    - GUI: Graphical user interface.
    - Canvas: Drawing area.
    - Simulator: Basically background of our program, the main piece.
    - File system: User files, library files, program files, printed files.

    - Scripting: Interpreter for quick interaction with program.
- Documents: Documents configuration item contains reports and documents related to our project.
    - Requirement analysis report
    - Initial design report
    - Final design report
    - Configuration management plan
    - Installation manual
    - User manual
- Baselines: Baseline configuration item is the milestones present in the project.
    - Requirement analysis
    - Final design
    - Prototype demo
    - Configuration management plan
    - Project implementation
    - Project testing
    - Project documentation
    - Project completion

3.2. Management and Control

This section is about the methodology for managing and controlling changes during the project development. This process is as follows:

Change Request: All team members may find out a need for change at any time during the lifetime of the project. This member must document his/her change request explaining the reason for change, list of relevant configuration items, and description of the change.

Analysis of Change Request: Requested change is examined by the team and team members approve the requested change if the change is found reasonable. If the requested change is found unnecessary, it is disapprove. Approved changes are passed to development team for embedding new changes in to implementation.

Implementation of Approved Changes: If the request for a configuration item is approved, development team, actually members involved with that task, takes over the approved changes. The development team checks the baseline CI and applies the approved change to the CI.

Version Control: After modification is done, new version of the product is named and released by CVS tool.

3.3. Configuration Status Accounting

We can say that Configuration Status Accounting (CSA) is recording and reporting of information needed to manage configuration effectively which is actually consists of a listing of the approved configuration identification, the status of proposed changes, deviations, and waivers to the configuration, the implementation status of approved changes and the configuration all units of the CI in the operational inventory.

CSA synchronizes team members during development process. But this information is not restricted with team members, assistant and instructors will also informed about project and they will be able to check out the project easily.

We are going to manage this through reports including all relevant information with the change, which member responsible for it prepares, and which will reside in related directory in CVS. Also we will use our web site and e-mails for accomplishing this task.

3.4. Auditing

Auditing is another essential element for this report and our project. It consists of independent examinations of work products and activities to assess compliance with designated criteria.

Software Configuration Audit (SCA) helps us to guarantee the changes are implemented in a suitable form. Auditing is done in order to be sure that the changes have been implemented correctly. We will do auditing regularly and we are going to perform it with increased frequency and details as the development proceeds.

Auditing will be accomplished by independent members of the group from the request. During auditing of a change request and its implementation, it will be checked whether the correct changes are made, any additional changes made, the changes are correctly

implemented and whether the final release is in compliance with the baseline and requirements. For finalization of each module, an auditing would take place.

4. Project Schedules – CM Milestones

We have divided our project into 5 modules. GUI, canvas, simulator, file system and scripting. These modules are also milestones and end of their implementations are also milestones for their CM activities.

	CM Milestones	
1.	CM Plan Submission	26-02-2006
2.	GUI audit	19-02-2006
3.	Canvas audit	26-02-2006
4.	Simulator audit	08-03-2006
5.	File system audit	02-03-2006
6.	Scripting audit	27-03-2006
7.	Documentation audit	15-04-2006

5. Project Resources

In our project we will use CVS provided by our department. And we will also use an ftp area for large files of or project. This will also be provided by the department.

6. Plan Optimization

As we have stated before our group is a small one and all members take part in all teams. We will have weekly meetings where we will discuss CM activities, accept or reject change requests and maintain CM plan. Then we will be applying changes during the following week.

One advantage of a being small group is we will be able to follow others process easily and meet more frequently when necessary.