Configuration Management Plan

by

ROYALFLUSH

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this Software Configuration Management (SCM) Plan is to identify, control, and report the changes to other team members, and to ensure accurate implementation. With the help of Configuration Management Plan, developers can work on the same project together, they can access old versions of the project and project members can realize changes that are done on the project. Due to the duration of the development phase of the project, some project materials can be changed or updated. When the CM is applied throughout the software development process, we will keep track of changes. In short, configuration management increases the quality of product, minimizes mistakes and helps the confusing and simultaneous changes to be controlled in a smart and definite way.

1.2 Scope of the Document

This document describes all configuration management activities employed for our project. This plan defines what will be put under configuration management control. It also includes who will be involved in the configuration management process. Moreover, this configuration management plan determines functions, responsibilities, timelines, various ways to handle changes in different versions, how software configuration management process can be implemented, and the development phases of the project.

1.3 Definitions, Abbreviations, and Acronyms

CI	Configuration Item
CM	Configuration Management
CSA	Configuration Status Accounting
CVS	Concurrent Versioning System
SCA	Software Configuration Audits
CMP	Configuration Management Plan
SCMP	Software Configuration Management Plan

CVS	Concurrent Versioning System
MAC	Media Access Control
IP	Internet Protocol
API	Application Programming Interface
GUI	Graphical User Interface
Wi-Fi	Wireless Fidelity
PIC	Peripheral Interface Controller
FTP	File Transfer Protocol
TFTP	Trivial File Transfer Protocol
PCB	Printed Circuit Board

1.4 Document References

- IEEE St. (IEEE Std 828-1998) for Software Configuration Management Plan
- http://www.cvc.org
- http://senior.ceng.metu.edu.tr/2008/royalflush

1.5 Document Overview

In the first chapter, purpose and scope of the document are introduced. We explain what CM is and reasons why it is essential. Moreover ,we describe some acronyms, definitions, and abbreviations .

In the second chapter, responsibilities are given to team members in the project. In this part of the CM, the organizational structure is explained in detail and the way members of the project communicate with each other is highlighted.

In the third chapter, configuration management, the subsections of identification, and configuration status accounting are described. CIs are identified.

In the fourth chapter, content and timelines of the project milestones are declared. The last two chapters of this document are devoted to project resources and timeline.

2. The CM Framework

2.1 Organization

Our team of developers is employing a maintainable plan to handle CM activities.

- Configuration Management Team (CMT)

  This team is mainly responsible for the maintenance of the configuration management plan and employ any changes to be made on it.

- Developer Team (DT)

  Our team of developers is in charge of applying market standards to the project to the extent of high productivity and maintainability. Members of this team are also encouraged to propose update plans to help testers comprehend the project requirements.

- Testing & Debugging Team (TDT)

  Core objective of this team is to maximize the efficiency of the final product by continuously testing and searching for possible defects at each phase of the project. Accordingly, creating change requests is another activity this team shall take on.

2.2 Responsibilities

Every member of our team is responsible for being compliant with ethical rules as well as market standards in order to help gather a robust application at the end. They are also expected to inform one another about possible defects and propose solutions for those that are maintainable.

Current versioning system (CVS) used extensively throughout the project must be comprehensible so that each member can check in, commit, and update records at any time they want. Nonetheless, prior to a substantial change in the code, members are supposed to inform each other about the possible changes and go on with the approval of the debugging team.

2.3 Tools & Infrastructure

The following tools will be used during the project.

- CVS

  CVS is a free, efficient, and easy to use version controlling system that enables developers to keep track of changes during a project regardless of the number of changes made to the original version of the source code. Restoring earlier versions is a crucial step to identify errors/flaws so this tool will be employed extensively during our project.

  .

- GNU C Compiler

  GNU C Compiler is an open-source, easy to use, platform independent, and a very efficient C compiler that is strongly recommended for low level developers thanks to its facility to generate readable code that helps debugging.

- GNU Debugger

  GNU Debugger is a high standard debugging tool used to test, analyze, and optimize the source code in a project. The software's great abilities ranging from listing the symbol table for the program being debugged to set breakpoints wherever one would like to during execution of it. Other than these, the amount of time to locate and correct the flaws/errors in the source code is no longer a concern for developers since a powerful counterpart is with them.

- Altium Designer

  Altium Designer is a user friendly software to make board designs as simply as possible. Besides CEng Embedded Board, we will be working on a custom made board and this program is of great help for us to locate possible flaws in the real-time scenario.

- Proteus Simulator

  Proteus Simulator is a high-performance parallel-architecture simulator that will be of help for us especially when group members work on their own, without having to connect AP-400 to their computers. This software supports LCD I/O; so, testing the programs we will be creating is not a concern. We believe it will be a facilitator for us during the project thanks to its high performance and real-time-aware facilities.

- TeraTerm

  TeraTerm is a developer-friendly, interactive, and customizable software that is capable of fetching the flash content of the device and listing it on a terminal window on 32-bit Win32 systems. Even though there are similar programs in the market, we decided to use this one because we will have to deal with command-line arguments and environment variables that are easily managed by the program of interest.

- CuteCom

  CuteCom can be regarded as a UNIX counterpart of TeraTerm since the two function exactly the same way as far as program deliverables are concerned. Advanced USB-port support puts this one step ahead of its rivals in the market. Since we will be working mainly on UNIX systems, this tool will be extensively used and taken help from.

3. The Configuration Management Process

3.1 Identification

In order to identify the current state of our project Wireless Network Traffic Monitoring System, it is mandatory for us to define a number of configuration items (CIs). We have chosen these items as CIs since they are the most crucial phenomena to concentrate on during the development of our project. Our CIs are:

- Code
- PCB
- Data Files
- Documentation
- Baselines

3.1.1 Code

Coding part is one of the most substantial parts of our project. There will be changes in the code at every stage of our project; yet there are a few standards about our source codes. These standards must be compatible with those of the main source codes of the device. They are:

- Variable names should be meaningful to be self explanatory.
- Code segments at each stage should be commented to be maintainable by another member of the project.

To be able to make the source codes concurrent, we will be employing the CVS routine as our basic CM tool.

3.1.2 PCB

As we are working on a real-time hardware project, one of the most important parts we have to make sure is the design and implementation of the Printable Circuit Board (PCB). During the design process, we will be taking the CEng336 Embedded board as our mainframe structure to build upon. We are using our graphical LCD displays and PIC16F877. The implementation and production of this board will also be executed by our team members.

3.1.3 Data Files

Data CIs consist of files for visualization of the menus on the LCD display. The events of the button configuration will also be stored in the data CIs.

3.1.4 Documentation

In order to make the final product robust and effective, we have documented every phase of our project. These documents are:

- Project Proposal
- Requirement Analaysis Report
- Initial Design Report
- Final Design Report
- Personal and General Weekly Progress Reports
- Configuration Management Plan
- Living Schedule

All of these documents are available under *Documents* section of our website.

3.1.5 Baselines

Baseline CIs will be the milestones and checkpoints of our living schedule. The main reason to identify these baselines is to maintain the project consistently. CVS tools will be used to define these baselines. These baselines are:

- Requirement Analysis
- Initial Design
- Detailed Design

- Prototype Development
- Configuration Management
- Design and Coding
- First Version Development
- Testing
- Documentation
- Final Demonstration

3.2 Management & Control

We have accomplished the necessary tasks of the project so as to maintain the project concurrently. Nevertheless, there may also be changes made by one of the members of the project on another member’s working area. To be more precise, at this part of the configuration management report, we are explaining the development, engineering, change request process and defect tracking.

3.2.1 Development

All of the issues related to our project are divided among the group members. Every member’s tasks and schedules are fixed. Our group member’s knowledge of the related issues about the project is equally likely. Therefore, group members are encouraged to help each other. When there is a problem that cannot be solved by a single member, all of the group members can work in conjunction.

3.2.2 Engineering

Since the beginning of the project, we have faced lots of technical and engineering problems. At the first step of the problem, we have tried to solve it as a group. Upon unsuccessful results, our assistant *Fatih Gökçe* and *the AirTies Company staff* were (and are going to be) helping us.

3.2.3 Change Request Process

As we mentioned earlier, during the development process, each member can find related issues with another member’s task; so, different members can make changes on different members’ works. In case such a change is in place, one member shall inform other member about the change request. When there is a pending request, all members will brainstorm and work on it to determine if the change is required. If that change really solves a problem and would be useful, than it will be integrated to the current version of the project.

3.2.4 Defect Tracking

As aforementioned, the last part of our project prior to the final demonstration is testing. So, at the testing stage, our group members,, our assistant, the company staff, and any other people interested in the scope of the project are welcome to test our product. If a problem occurs at this stage, our team members will work on and try to correct it before the final demonstration takes place.

3.3 Configuration Status Accounting

We have related information about configuration status in our website. It includes all of our documents and living schedule. Reports like Build Reports, Defect Reports, Release Reports will also be on our website. We will start to use CVS system as soon as possible to be able to give versioning information.

We paid for the materials used since the beginning of the project ourselves.

3.4 Auditing

To be able to develop a complete and standardized project, auditing is a remarkable aspect. We have been making the auditing process regularly. By means of this process, we can show possible faults before they occur and see the nuisances and possible effects of the changes that were made. Auditing process is mainly consisting of three parts: Functional Audits, Physical Audits and Process Audits.

3.4.1 Functional Audits

The main aim of functional audits is mainly checking the functionality and performance of the configuration items. Another one is checking the consistency between the design specifications and products.

3.4.2 Physical Audits

The aim of the physical audits is a formal examination of the existing configuration items to verify the requirements, specifications and standards applied in producing the configuration items.

3.4.3 Process Audits

The aim of the process audits is checking whether the process consistently produces acceptable results or not. The process stage is continuous, so we need to do regular process audits from time to time.

4. Project Schedule

The living schedule we have prepared clearly identifies all the tasks that need to be done and remarkable deadlines that need to be met throughout this semester. All group members will follow this schedule while working on the tasks assigned to them. The living schedule published on our project website will be updated regularly. Core milestones and respective due dates are listed below.

	CM Milestone	**Due date**
1	CM Plan Delivery	09.03.2008
2	1st Development Snapshot Demonstration	13.03.2008
3	Version I	27.03.2008
4	Testing	17.04.2008
5	Version II	24.04.2008
6	Final Package + Demonstration	13.06.2008

5. Project Resources

In order to apply our CM plan and folow the CM activities, we will use CVS extensively. Our team consists of four people and it is crucial to work on the source code without inconsistencies. With the help of CVS, updates and changes to the source code are handled easily and new versions are produced without losing the older versions.

We are fully aware of the fact that various technical and human resources are consumed during the project. Thus, we are trying to complete the project within milestone deadlines in order not to lead to any unnecessary work and effort.

6. Update Plan

The configuration management plan will be a beneficial guide for our team in our project. Yet, there may be a number of changes during the development process and in such cases we will make necessary changes and/or updates in our configuration management plan and in our recent works. Communication between the team members is an important issue in order for all to be aware of the changes and easily follow the plan. All team members will be responsible for being notified of the updates and changes in the system, so incompatibilities and possible delays are likely to be eliminated..