



MIDDLE EAST TECHNICAL UNIVERSITY



DEPARTMENT OF COMPUTER ENGINEERING

CENG 492 - SENIOR PROJECT

CONFIGURATION MANAGEMENT PLAN

Version <1.0>

09.03.2008

PREPARED BY



Aycan Tekerek

1347988

aycantekerek@gmail.com

Barış Yanar

1348093

barisyanar@gmail.com

Şeniz Yıldırım

1502814

senizyildirim@gmail.com

Wai Phyoe Maung

1368687

waiphyoe2003@gmail.com

Revision Sheet

Release No.	Date	Revision Description
Rev. 1.0	9/03/2008	Configuration Management Plan

TABLE OF CONTENTS

	Page #
1.0 INTRODUCTION	4
1.1 Purpose	4
1.2 Scope	4
1.3 Audience	4
1.4 Definitions, Acronyms and Abbreviations	4
1.5 Document References	5
1.6 Document Overview	5
2.0 THE ORGANIZATION CM FRAMEWORK	6
2.1 Organization	6
2.2 Roles and Responsibilities	6
2.3 Tools and Infrastructure	7
2.3.1 CVS	7
2.3.2 WebMethods Integration Server	8
2.3.3 SoftwareAG Crossvision Application Designer	8
3.0 THE CM PROCESS	8
3.1 Identification	8
3.2 Configuration Management and Control	9
3.3 Configuration Status Accounting (CSA)	10
3.4 Configuration Audits	10
4.0 PROJECT SCHEDULE – CM MILESTONES	11
4.1 Project Schedule	11
4.1 CM Milestones	11
5.0 PROJECT RESOURCES	11
6.0 PLAN OPTIMIZATION	11

1.0 INTRODUCTION

1.1 Purpose

Software configuration management is a set of methods and tools to ensure the integrity, traceability and modifiability of products created throughout the life cycle of a software development project. In scope of this definition, the purpose of this CMP is to ensure that the suitable procedures on changes and version control are clearly stated and followed by all team members. The entire development team should be familiar with the contents of this document.

In particular it provides information about:

- The sequence of steps to systematically make changes to developed software.
- The organization and documentation of the project.
- Naming convention for designated parts and releases.
- How to control versions of software.
- Tools used to develop project.

1.2 Scope

This document explains all details of Configuration Management (CM) process of ISTEIS project being developed by Taseron Yazilim. This CMP is applicable to all work performed as part of the ISTEIS project, which includes the design, implementation, integration and testing of the modules of the ISTEIS. This document encompasses the lifecycle of the project. It provides the basis for all configuration management activities in support of ISTEIS project and describes responsibilities of team members in this process. Configuration management (CM) requirements, methods, basic procedures and requirements necessary to provide configuration control of the documents and the software, and also the accurate reporting of configuration status are also clearly stated in this CMP.

1.3 Audience

The intended audience for this document is the members of Taseron Yazilim, our supporting company, our project supervisor and instructor.

1.4 Definitions, Acronyms and Abbreviations

ACRONYM	DEFINITION
CCB	Configuration Control Board
CI	Configuration Item
CM	Configuration Management
CMP	Configuration Management Plan
CSA	Configuration Status Accounting

CVS	Concurrent Version System
SCM	Software Configuration Management
SCMP	Software Configuration Management Plan
SCR	System Change Request

Configuration Control Board represents the group of people who accept or reject the proposed changes.

Configuration Items are the things to which configuration management can be applied such as the documents describing the development of the product, or modules.

Auditing describes the process of determining whether the developed parts of the product satisfy all the requirements or not.

Configuration Control represents the process of request, evaluation, approval, and implementation of changes to configuration items.

Baselines are used as a reference during the lifecycle of the project such as the design reports, snapshots, and demos.

1.5 Document References

The references used to prepare this CMP for our project ISTEIS can be listed as following;

- Pressman, Roger S. (2001). *Software Engineering: A Practitioner's Approach*, Fifth Edition. New York, NY: McGraw-Hill.
- IEEE Standard for Software Configuration Management Plans—IEEE Std 828-1998.
- IEEE Guide to Software Configuration Management—ANSI/IEEE Std 1042-1987.

Also this document has a compliance with our previous documents of the ISTEIS project such as Requirement Analysis Report, Initial and Final Design Reports.

1.6 Document Overview

Our CMP consists of 6 sections. Below you can find the name of these sections and their brief explanations

- **Introduction:** In this part, the purpose of CMP and the scope of the document are stated. Moreover, the acronyms, abbreviations used in this document and their definitions are clearly explained. Lastly, the references taken as guidelines to prepare this SCMP are given.
- **The Organizations CM Framework:** The organization and responsibilities of each team member for CM are clearly stated. Also, the infrastructures and the tools being used throughout the project are introduced.

- **The CM Process:** In this part, we examined the configuration management process. The identification of the CIs is made and then we specified the management, control and audit of these CIs.
- **Project Schedule – CM Milestones:** In this part, schedule for configuration management activities is determined and we stated the milestones and deadlines respectively.
- **Project Resources:** In this part, we explained the Project resources which will be necessary for CM.
- **Plan Optimization:** In this part, we examined the cases to make optimizations on our CMP and how to realize these optimizations.

2.0 THE ORGANIZATIONS CM FRAMEWORK

2.1 Organization

In order to apply CMP effectively, organizational units taking part in CM activities should be properly determined and described. Taseron Yazilim has a hierarchical organization in CM Framework. The top organization on this framework is the Configuration Control Board (CCB), which handles the control of CM activities. As our project is developed by a small group of people, just four people, every member of our team is on the Configuration Control Board. The other organizational units under CCB are listed below:

- Configuration Management Team (CMT)
- Developing Team (DT)
- Testing and Debugging Team (TDT)
- Release Control Team (RCT)

These 4 units are in contact with each other and each team member also participates in all these units.

2.2 Roles and Responsibilities

The responsibilities of the units mentioned above are as follows:

Configuration Control Board (CCB)

- Reviews SCRs offered by testing and debugging team and analyzes the effects of the changes,
- Accepts/rejects the SCRs,
- Holds audits,
- Reduces the negative effects of changes made.

Configuration Management Team (CMT)

- Creates and maintains CMP,
- Coordinates CMP activities and ensures that the activities are implemented on time,
- According to the works done, updates CM schedule,

- Reports the changes to other units.

Developing Team (DT)

- Main responsibility of CMT is to implement the project source code,
- According to the SCRs, makes changes to the parts that are mistaken and implements again.

Testing and Debugging Team (TDT)

- Main responsibility is testing and debugging the source code,
- According to the result of testing process, offers SCR.

Release Control Team (RCT)

- Main responsibility is to control the releases of the project,
- Creates baselines.

Moreover, the general rules for each member of Taseron Yazilim to apply and maintain the configuration management plan is listed below as:

- Performing assigned CM activities.
- Completing the assigned jobs until deadline to stick to the CM schedule.
- Informing the people about an SCR by sending the proper e-mail.
- When some change is done in CVS, informing all the members via our mail list.
- Depending on the CM process, checking in and checking out by using CVS properly.
- Commenting on the changes properly, while committing to CVS.

2.3 Tools and Infrastructure

2.3.1 CVS

CVS will be our project's version control system. CVS allows each team member to check out resources (source code, document etc.) upon which he/she plans to work and also to check these resources back into repository when they are finished. Different from the other similar applications CVS does not use a locking strategy but an optimistic merge strategy which allows it to deal with potential conflicts. It also provides extensive logging, auditing, and tracking facilities. If some new bugs appear after the change is made with the help of CVS, old versions of the project will be retrieved easily to find out which change caused the bug. The CVS repository stores a complete copy of all the files and directories which are under version control. When a set of changes has been finished, they are committed. The repository then contains the changes which have been made, as well as recording exactly what has been changed and when it was changed.

2.3.2 WebMethods Integration Server

WebMethods Integration Server is a java based, multi-platform enterprise integration engine that supports the execution of services to perform the integration logic such as data mapping and communication with other systems.

2.3.3 SoftwareAG Crossvision Application Designer

Crossvision Application Designer is a powerful design-time and run-time environment that uses AJAX to create sophisticated Web-browser interfaces as part of a Service Oriented Architecture. It accelerates and simplifies the design and deployment of rich Internet Applications (RIA) that use the AJAX pattern (Asynchronous JavaScript + XML).

3.0 THE CM PROCESS

3.1 Identification

The 'ISTEIS' project can be identified as a combination of the following configuration items (CIs): Job Seeker Modules, Job Provider Modules, Administrator Modules, Documentation and Baselines.

- ***Job Seeker Modules***

These modules include the GUI and web services to be developed for the job seeker users of our web portal. These modules are namely; Job Search, Job Application, Information and Feedback Modules that we have defined in our previous documentation.

- ***Job Provider Modules***

These modules include the GUI and web services to be developed for the job provider users of our web portal. These modules are namely; Payment, Career Site Selection, Employee Search and Information Modules that we have defined in our previous documentation.

- ***Administrator Modules***

These modules include the GUI and services to be developed for system administrators to manage our portal.

- ***Documentation***

The ISTEIS project has a detailed documentation that can be referred efficiently in case of any changes occurred in the modules and can be easily updated after a module configuration.

The following reports are prepared so far:

- Project Proposal
- Requirement Analysis Report
- Initial Design Report
- Detailed Design Report

➤ Configuration Management Plan (CMP)

- **Baselines**

Throughout the project we have our milestones to consider. Before starting to implement ISTEIS, we have prepared some documents to collect and clearly state the necessary information for further steps in development. Each report and milestone provided us to see lacking parts of our modules, reconsider their interactions, data management and improve our project with new features and different designs for modules.

Following list constitutes milestones for ISTEIS that we have completed or that we will consider in the later steps:

- Proposal of ISTEIS
- Requirements Analysis
- Initial Design
- Detailed Design
- Developing Prototype
- Configuration Management Plan
- Implementation
- Module Interactions
- Testing
- User Manual
- Installation Plan

3.2 Configuration Management and Control

Change Request: Any change requests will be mainly held via e-mail to the group mail list and will be discussed during our weekly meetings. Requests can only be made for a previously completed sub-module. The change requests should have the following information:

1. Title of proposed change.
2. Affected CI(s).
3. Originator's name.
4. Date of request.
5. Date of proposed implementation.
6. Indication of urgency for change approval.
7. Description of the proposed change.
8. Justification for the proposed change.
9. Estimated schedule impact on other members and whole project.
10. Estimated performance impact on other components or modules of the project.
11. Any Additional Comments.

Change Request Evaluation: For evaluation of change requests our baselines listed in section 3.1 will be the reference. The group members are informed with the CR by e-mail sent to the group mail list. As taking the information given in the change request all members make their own investigations about the change, how can it be done, how it can affect their progress or whole project. After these individual investigations team discusses the request in following weekly meeting. If an approval results in CCB, the group member assigned to task that needs to be changed takes the responsibility to complete the change by the date determined in CCB meeting. If the CR is disapproved by the team, then the change originator team member also reports this result of the evaluation process. If there are more than one CR to be discussed in a meeting, the discussions are made in the order of urgency levels of the CRs.

Change Implementation: Once the change request has been approved, the change is implemented by the developer who has initially implemented the changed part. CVS will be used to make the necessary changes. Once the change has been tested totally in that environment, it will be integrated to the completed system parts. The related documentation changes should also be done by the responsible group member. Module or part version, name and release date will be used to track the configuration items. This tracking will be done by the whole team.

3.3 Configuration Status Accounting

As a software project group, our team needs a formal and well defined way to record and check the information about the changes that have been made. The configuration status of all designs and changes will be communicated to all team members. We have weekly formal meetings and group mail list, any change notification can be occur in these ways. If a member in our group decides on a change, firstly it has to be discussed as a whole group having the authority of CCB. If CCB agrees on the specified change, it is committed in below described manner by the members who are responsible from the related task or module:

- Include very detailed report about why modification is needed,
- Put the modified source code in related part of CVS,
- Check whole project again for preventing conflicts and system errors,
- Provide the group members with a detailed documentation of how and when the module is modified.

3.4 Configuration Audits

Configuration auditing is an important step in developing software projects. Although we are working as a small group of four people, our project team still needs a way to maintain integrity among different modules of our project. Although our design documentations shape our implementation, we still can not know what we may face during implementation phases. Any subtle change can cause differences in functionalities. Therefore, every member of our team has to demonstrate the functionalities he/she implemented, state the problems he/she faced and list the changes he/she had made in functionalities that are agreed by CCB before. In this process, other group members have the functionality of controllers. We think that in this way, incompatibilities between modules o parts of our software can be decreased to minimum level.

We will have functional, physical and process audits. Functional audits will be used whenever a new version of product is released. These audits will mainly check functionalities are parallel with the ones defined in the project documentation. We will also have physical audits, which will be conducted before our final release. These audits will have more coverage and will also check if there are missing functionalities.

4.0 PROJECT SCHEDULE – CM MILESTONES

4.1 Project Schedule

Our living schedule clearly identifies all tasks needed to be done throughout the project and the milestones determined for this semester. We will be progressing according to this schedule. Every group member has been assigned to group of tasks for each week. On each of our weekly meetings, each member will inform their progress on the assigned tasks to other members so we can make sure of that the living schedule is being followed. The living schedule located on our web site will be updated according to our progress on regular basis.

4.2 CM Milestones

The followings are our CM Milestones:

CMP Delivery	09/03/2008
Job Seeker Modules Audit	16/03/2008
Job Provider Modules Audit	06/04/2008
Administrator Modules Audit	20/04/2008

5.0 PROJECT RESOURCES

In order to keep track of changes done on our project we will use CVS. CVS helps us providing consistency on each source that every member of our team works, so the newer versions of the project can be produced without losing the older versions.

To easily apply CMP and follow CM activities, we have all our project documents and living schedule which we are updating according to our progress regularly on our web site.

Throughout the development of our project, we will be using WebMethods Integration Server and SoftwareAG Application Designer tools which are provided by our supporting company in this project, SoftwareAG.

Since all members of our team have their own and shared responsibilities on CMP, Taseron Yazilim constructs the most important part of the project resources as human resources.

6.0 PLAN OPTIMIZATION

This CMP document will be a guide in the development of our project ISTEIS. Since we have a small team consisting of four people and we are having weekly team and project mentor meetings, it seems that we do not need much optimization on our CMP. However, there can be some changes during the development process and in such cases we should

make necessary optimizations in our configuration management plan and in our progress. At this point, communication between team members becomes crucial. Our group mail list is also used for this purpose. Any team member wants or does a change he/she must inform the other team members via sending an e-mail describing the corresponding change to this mail list. Moreover, there will be a file in CVS system where all team members add the latest updates and notes on the project in order to give information and we will discuss these updates during our weekly meetings. By this way, we will be able to keep track of changes and delays in our progress schedule and make our plan optimization frequently.