

Cansu Hekim – 1630888
Bekir Doğru – 1560085
Zaman Safari - 1572254
Hamit Siddiqi - 1634427

Siemens EC HRMS

Configuration Management
Plan Report

26.03.2012

Table of Contents

1. INTRODUCTION.....	2
1.1. PURPOSE OF CONFIGURATION MANAGEMENT PLAN	2
1.2. SCOPE OF THE DOCUMENT	2
1.3. DEFINITIONS, ACRONYMS AND ABBREVIATIONS.....	3
1.4. DOCUMENT REFERENCES	4
1.5. DOCUMENT OVERVIEW	4
2. THE ORGANIZATIONS CM FRAMEWORK	5
2.1. ORGANIZATION	5
2.1.1. <i>Software Development Team (SDT)</i>	5
2.1.2. <i>Configuration Management Team (CMT)</i>	5
2.1.3. <i>Change Control Team (CCT)</i>	5
2.1.4. <i>Testing Team (TT)</i>	5
2.1.5. <i>Release Control Team(RCT)</i>	5
2.2. RESPONSIBILITIES.....	6
2.3. TOOLS AND INFRASTRUCTURE	6
3. CONFIGURATION MANAGEMENT PROCESS	7
3.1. IDENTIFICATION.....	7
3.1.1. <i>Source Code</i>	7
3.1.1.1 Java Package.....	7
3.1.1.1.1 pkg model:.....	7
3.1.1.1.2 pkg dao:.....	7
3.1.1.1.3 pkg controller:	8
3.1.1.1.4 pkg util:.....	8
3.1.1.1.5 pkg session;.....	8
3.1.1.2 Web Content	8
3.1.2. <i>Data</i>	9
3.1.3. <i>Documentation</i>	9
3.2. CONFIGURATION MANAGEMENT AND CONTROL.....	9
3.2.1. <i>System Change Requests</i>	9
3.2.2 <i>System Change Evaluation</i>	10
3.2.3. <i>System Change Implementation</i>	10
3.3. CONFIGURATION STATUS ACCOUNTING	10
3.4 AUDITING	10
4. PROJECT SCHEDULES AND CM MILESTONES.....	11
5. PROJECT RESOURCES	11
6. PLAN OPTIMIZATION	12

1. Introduction

1.1. Purpose of Configuration Management Plan

It is well known that, companies develop larger software to satisfy the needs of the technologies. However, in addition to the implementation of a software, codes reliability, maintainability, the ease of the update operation according to the improvement in technology and of course the ease on the testing of the code (i.e. convenience on unit testing). These specialties require some specific design patterns, documentation, keeping track of the process and the updates. At this point, configuration management report give the basic information about the process, design pattern/s used, and how to implement updates with the/ these design pattern/s. When a change required, the questions of who will do it, how he/she will do it and how it guarantees that the change made won't cause some other problems blinks on the minds [1]. With the help of the configuration management report these questions are well answered.

Since HRMS, which is sponsored by Siemens EC, is a project having a huge database, huge number of classes mapping to each database table, and the codes of the ZK framework basically implementing the user interfaces, it is hard to organize the updates and how these will be handled. Therefore, it is a need for our team to improve our project' s maintainability and reliability and to ease the handling of the processes mentioned above. Moreover, since our project is an intranet project having a user interface, it requires some specific design patterns like model-view-controller, with the help of the CM report we will be able to provide the stability of our project HRMS.

1.2. Scope of the Document

The scope of this document is the HRMS project, implemented by the group InnovaSoft and sponsored by Siemens EC. This document gives information about the timeline of our implementation process a point of view about the design of the implementation, the organization of the database and their maintainable aspects (i.e. future need of an introduction of a new role to a system). Moreover, this document specifies organization of our team and the responsibilities of each team member also responsibilities as subgroups etc. The tools that we are used are take place in this document in order to ease the modification of the project when needed.

The intended audience of this document is the developer team, InnovaSoft, the sponsor company, SIEMENS EC, and the instructors responsible for CENG492 course at Computer Engineering Department of Middle East Technical University.

1.3. Definitions, Acronyms and Abbreviations

Abbreviations	Definitions
HRMS	Human Resource Management System
SIEMENS EC	Siemens Enterprise Communication
CENG	Computer Engineering
CM	Configuration Management
CMP	Configuration Management Plan
SDT	Software Development Team
CMT	Configuration Management Team
CCT	Change Control Team
TT	Testing Team
RCT	Release Control Team
SCR	Specification Change Request
SVN	Subversion
SRS	Software Requirements Specifications
IDR	Initial Design Report
DDR	Detailed Design Report
CI	Configuration Items
pkg	Package
dao	Data Access Object
pdm	Personal Data Module
pem	Performance Evaluation Module

Table 1 : Abbreviations

1.4. Document References

References	
[1]	Configuration Management, The presentation prepared by CENG492 instructors
[2]	http://en.wikipedia.org/wiki/Eclipse_(software) Sec. 8.5
[3]	http://en.wikipedia.org/wiki/Subversion_(software)
[4]	http://en.wikipedia.org/wiki/Trac
[5]	http://en.wikipedia.org/wiki/ZK_(framework)
[6]	IEEE Standard for Software Configuration Management Plans (IEEE Std 828-2005)
[7]	http://senior.ceng.metu.edu.tr/2012/innovasoft/

Table 2: References

1.5. Document Overview

This document has six main sections namely Introduction, The Organizations CM Framework, Configuration Management Process, Project Schedules and CM Milestones, Project Resources, Plan Optimization.

In the introduction part what is CM report and why it is needed is described also the scope and the overview of the document are mentioned.

In the organizations CM framework part, the organization of the team and the responsibilities of each team member in order to sustain maintainability and implement the CMP to make our project reliable and maintainable, are described.

In the section of Configuration Management Process after mentioning about the identification of the project (i.e. source code, data, and documentation of the project), configuration management process is described in the subtitles Configuration Management and Control, Configuration Status Accounting, Auditing.

In the part of Project Schedules and CM milestones, the timeline that we followed while processing the CM, and the basic main points that we used in our design may be implemented in order to update the project according to the needs of the company.

In the project resources part, basic resources that we used in the process of the documentation and implementation of our project, HRMS, are described.

In the plan optimization part, the optimization of the configuration and progress of our project, HRMS, is analyzed.

2. The Organizations CM Framework

2.1. Organization

Our team consists of four members. These members are Bekir DOGRU, Hamid SIDDIQI, Cansu HEKIM, Zaman SAFARI. In order to have a qualified and successful project, all of the team members take part in the CM. As a result the members shared different tasks and arranged sub-teams which have different missions. The sub-teams and their tasks are explained below:

2.1.1. Software Development Team (SDT)

Responsible Members: ALL

The responsibility of SDT is to implement the modules of the HRMS. This team integrates the implementations with the rest of the parts. Moreover, when the testing team (TT) requests changes, they are made by SDT. This team has other responsibilities such as fixing bugs, creating releases and updating the source code via SVN.

2.1.2. Configuration Management Team (CMT)

Responsible Members: Bekir DOGRU and Cansu HEKIM

Configuration Management Team (CMT) is mainly responsible to maintain the organization of CM. In order to sustain the organization, CMT keeps the plan up-to-date.

2.1.3. Change Control Team (CCT)

Responsible Members: Zaman SAFARI and Hamid SIDDIQI

The main responsibility of CCT is to accept or reject SCR's, to review SCR's and to monitor SCR's. Furthermore, this team supervises all the activities of the other groups.

2.1.4. Testing Team (TT)

Responsible Members: Bekir DOGRU and Zaman SAFARI

The Testing Team (TT) is responsible for testing and debugging of the implementations produced by SDT. Meanwhile, TT takes the responsibility to control adequacy of the modules. If the requirements are not fulfilled, this team may want to change requests. After testing, this team gives feedback about the modules.

2.1.5. Release Control Team(RCT)

Responsible Members : Cansu HEKIM and Hamid SIDDIQI

This team will control the current and next versions of the HRMS. Also this team is responsible to merge different branches of HRMS.

2.2. Responsibilities

Although team members have different responsibilities in different sub-teams namely SDT, CMT, CCT, TT and RCT as explained in Section 2.1, all the members of HRMS take part in Configuration Control Board (CCB) so they share same individual responsibilities. Some of those critical responsibilities are mentioned in the bottom lines:

- When a change is made on the source code by a member, it should be commented before it's put into repository through SVN.
- Each member should follow the pre-defined CM schedule.
- Other members should be informed about SCR via e-mail.

2.3. Tools and Infrastructure

- **Eclipse**

For development IDE, we will use Eclipse. Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. It is written mostly in Java and can be used to develop applications in Java and, by means of various plug-ins, other programming languages including Ada, C, C++, COBOL, Perl, PHP, Python, R, Ruby (including Ruby on Rails framework), Scala, Clojure, Groovy and Scheme.[2]

- **SVN (Subversion):**

Apache Subversion (often abbreviated SVN, after the command name svn) is a software versioning and a revision control system founded and sponsored in 2000 by CollabNet Inc. Developers use Subversion to maintain current and historical versions of files such as source code, web pages, and documentation. Its goal is to be a mostly-compatible successor to the widely used Concurrent Versions System (CVS)[3]. SVN maintains the current and old versions of source codes, web pages and documentation on the server. In our project, we will use Eclipse's plug-in (Subclipse) to commit or update the source code.

- **TRAC:**

Trac is an open source, web-based project management and bug-tracking tool. The program is inspired by CVSTrac, and was originally named svntrac due to its ability to interface with Subversion.[3] It is developed and maintained by Edgewall Software[4]. With the help of the

TRAC, we will assign tasks to the members of SDT and also SCR's are raised. Every member of the team has to learn using Trac. Each member is going to use Trac in order to show his progress.

- **ZK Framework**

ZK is an open-source Ajax Web application framework, written in Java, that enables creation of rich graphical user interfaces for Web applications with no JavaScript (despite the J in Ajax acronym standing for JavaScript) and little programming knowledge. The core of ZK consists of an Ajax-based event-driven mechanism, over 123 XUL and 83 XHTML-based components, and a markup language for designing user interfaces. We use this framework within Eclipse IDE to design our interfaces.[5]

- **Webpage**

All of the documents, information about the project and the project progress can be seen on our web page.[7]

3. Configuration Management Process

3.1. Identification

The Configuration Items (CI) can be separated into 3 categories namely source code, data, and documentation, which are explained in detail as follows.

3.1.1. Source Code

Source code of the HRMS, as an intranet project, can be analyzed in two parts namely java packages and web content. Our java packages are models, dao, controller, util, session and web content is consisted of zul files, which are described in detail as follows.

3.1.1.1 Java Package

3.1.1.1.1 pkg model:

This package contains the classes mapping to each database table.

3.1.1.1.2 pkg dao:

Each class in the model package has a dao class, which accesses the database without querying the database.

dao.admin: This package contains the dao classes which accesses the database tables related to A&A module.

dao.pdm: This package contains the dao classes which accesses the database tables related to Personal Data Module.

dao.pem: This package contains the dao classes which are related to the Performance Evaluation Module.

3.1.1.1.3 pkg controller:

This package is responsible for the user interface i.e. what can be seen by each role, editable, updatable fields, button functions etc.

controller.admin: This package contains the controller classes which are responsible for the user interface specifications of the A&A module.

controller.pdm: This package contains the controller classes which are responsible for the user interface specifications of the Personal Data Module.

controller.pem: This package contains the classes which are responsible for user interface specifications of the Performance Evaluation Module.

3.1.1.1.4 pkg util:

Package util contains the class of hibernate util. It is auto generated and configures the database transaction. dao classes which are responsible for access of the database use this class to do their work.

3.1.1.1.5 pkg session:

Package session contains only one class, user.java, which configures the information of the current user (his/her role, and the authentications). Controllers are getting information from this class to do their work.

3.1.1.2 Web Content

Web content is constituted of the zul files, which are the html like files. They are the files, which are used in interface design.

3.1.2.Data

HRMS uses a database, which is organized in the way that can be half generic meaning we can define and introduce new roles to the system. It is well organized according to the needs of the company. They are basically holds the data about personal information and performance evaluation items and A&A items.

The database tables can be divided into three parts:

1. tblappl (table application)
2. tblemployee (table employee)
3. tblpem (table personal evaluation module)

3.1.3.Documentation

All documents are written with the contribution of all team members and most of them can be found in our website. The documents written so far includes:

- Project Proposal
- Software Requirements Specifications
- Initial Design Report
- Detailed Design Report

3.2. Configuration Management and Control

3.2.1. System Change Requests

Minor System Change Requests are directly added to the system and handled by SVN and requires no extra information. However, major changes are controlled by the Trac system. In this system a System Change Requests should consist of

- Team member name
- Description
- Date
- Deadline
- Related Category/Module
- Priority and Version Number

When the request is made, a ticket is opened by Trac and it can be seen by all members of our team.

3.2.2 System Change Evaluation

The main discussions about the evaluation of the SCR are done by tickets in the Trac System and their logs are kept. However, also in the face to face meetings, the evaluation process of the SCR's are discussed. Members can give opinion about the SCR and the best way of evaluation will be determined.

3.2.3. System Change Implementation

When the change is accepted by all the members, the followings are done. Firstly, the source code is changed as required. Then, unit testing is done. Lastly, the changes in the source code is updated in SVN.

3.3. Configuration Status Accounting

We have introduced configuration items in previous sections and the information regarding configuration items is needed to be stored because the control of configuration items gets complicated as more changes are made during development. Since keeping track of development process of the project is one of the most essential factors to be handled, we will use different ways to express these changes and updates at the same time.

The information will consist configuration identifications, change the information of request and information about the details of the implementation. While approaching to the end, comments of the SVN commits and meeting reports will guide us through the common changes. Also, versioning of the project will be controlled by comments and defining the use of updates.

Finally, information about the development process of the project and the problems and their solutions will be published by our web page.

3.4 Auditing

All the members will take responsibility for accomplishing the task of auditing. Auditing is regularly done by weekly meetings and according to the iteration schedule we have already prepared. In the mean time changes that are made on a CI will be checked during the auditing phase. Auditing of the code will be done weekly by using appropriate test methods. Also, each team member should check his/her own part of the code to determine its correctness. Each member can commit that code to SVN after self-checking.

Meanwhile, before the commit, each member should make sure that code has compiled and working properly. By checking this and committing the code after this test, team can be sure that the project that is kept in the SVN can be compiled and working correctly.

No to mention, project schedule should be checked and updated regularly in order to obey the timing that is planned.

4. Project Schedules and CM Milestones

Our project milestones are determined according to the restrictions in the course schedule.

September 2011 – January 2012 are the period of literature research, meeting with Siemens EC about the tools that we are going to use to develop the HRMS and the interface design, and the documentation. In this period we completed the Project Proposal, SRS, IDR, DDR.

In January 2012: After getting information about hibernate and Zk framework, we completed our very first demo and presented.

Project Milestone	Date
Revised Design Report	21 / 02 / 2012
Implementation of database	10 / 03 / 2012
Web Page Publish	25 / 03 / 2012
CM Plan Report	25 / 03 / 2012
Committing the code to SVN	30 / 03 / 2012
Implementation of HRMS	30 / 03 / 2012
Integrations of Units and Testing-1	03 / 04 / 2012
First Development Snapshot	06 / 04 / 2012
Test Specification Report	17 / 04 / 2012
First Release / Demo	15 / 05 / 2012
Final Package/User Manual	22 / 05 / 2012

Table 3 :Project Milestones

5. Project Resources

The resources of the HRMS project can be listed as follows:

- SVN (Subversion): Source code management and versioning tool.
- Trac: Issue Tracking System.
- Project website: Information and news about the project.

- MSQL : Microsoft SQL Server
- Eclipse: Development IDE
- ZK framework : Leading Enterprise Java Web Framework

6. Plan Optimization

The HRMS project's development will be done in accordance with the CMP. Configuration Control Board is going to control all the updates and changes in this report. If any change or update comes into this document, TRAC system will warn all the users. Meanwhile, The attendance for the weekly meetings among group members is vital and a must. According to these meetings and iteration schedule the tasks are going to be performed regularly and be published in the web site of the group. By keeping the Configuration Report valid, we are able to monitor the development status while working for the implementation. All of our group members are responsible for updating and editing this report. A plan optimization can occur when an update occurs in this report. Not to mention that, weekly meetings with our assistants guarantees that we move according to our plan.