

CENG 492

Computer Engineering Design II



Configuration Management Report for Bookwiser

Ballmer Peak
Elif Alev AHMET
Betül DOLAY
Batuhan KARAGÖZ
Ozan TABAK

March, 2011

Contents

1. Introduction	3
1.1. Purpose of Configuration Management Plan	3
1.2. Scope of the Document	3
1.3. Definitions, Acronyms and Abbreviations	4
1.4. Document References	4
1.5. Document Overview	5
2. The Organizations CM Framework	5
2.1. Organization	5
2.2. Responsibilities	6
2.3. Tools and Infrastructure	6
3. Configuration Management Process	7
3.1. Identification	7
3.1.1. Source Code	7
3.1.2. Data	7
3.1.3. Documentation	7
3.2. Configuration Management and Control	8
3.2.1-System Change Requests	8
3.2.2-System Change Evaluation	8
3.2.3-System Change Implementation	8
3.3. Configuration Status Accounting	8
3.4. Auditing	9
4. Project Schedules and CM Milestones	9
5. Project Resources	9
6. Plan Optimization	10

1 Introduction of Configuration Management Plan

1.1 Purpose

The purpose of this document is to identify and describe the configuration management (CM) process for the Bookwiser project. It aims to provide an overview of the organization, activities, overall tasks, and objectives of Configuration Management for the project. It addresses configuration item (CI) identification, control and configuration audits at a high level. Besides, additional details regarding CM activities, techniques, and tools are provided.

Configuration Management Plans (CMP) are prepared as a methodology for handling some situations in the project which are caused because of changes and updates. For software developers, it is also important to realize that applying configuration management really increases quality of the product. Benefits of CMP include the stability of design and implementation of all components and subsystems so that interfaces between subsystems are reliable and accurate. When changes are needed, CMP will facilitate and ensure that appropriate communications occur among related developers and all who are potentially affected by the change. Participants are assured to be informed of requested change, and they will have opportunity to take a part in discussion of the changes scope and impact. After change is decided to be made, participants can modify their subsystems design as needed.

In summary, CMP helps developers control and identify the causes and results of changes in a software development process so that these all can be used in the implementation of changes. Guidance of a CMP is essential for all software teams' communication and coordination on a project.

1.2 Scope of The Document

The scope of this document is about the identification of configuration management plan for project Bookwiser System Project of Ballmer Peak software development team. CMP presents configuration management activities that will be applied throughout the development process of our project and these activities explained in this document are applicable during the development and maintenance phases of the project.

The document contains methodologies and activities designed to manage changes and

relate them. Also tailoring, differencing work product and reporting on all are included in this document too. Because the project includes a medium number of modules, includes some integrated libraries and is being developed by a group of five people. The related audience for this document is Ballmer Peak software team, our assistant and instructors.

1.3 Definitions, Acronyms and Abbreviations

CM – Configuration Management

SCR – Specification Change Request

SVN – Subversion

SDT – Software Development Team

TT – Testing Team

CCT – Configuration Control Team

CMT – Configuration Management Team

CI – Configuration Item

1.4 Document References

1. Pressman, Roger S. (2001). Software Engineering: A Practitioner' s Approach, Fifth edition. New York, NY: McGraw-Hill

2. IEEE Standard for Software Configuration Management Plans (IEEE Std 828-2005)

3. A. Dogru, Component Oriented Software Engineering, The Atlas Publishing, Turkey, 2006.

4. Software Configuration Management?, The presentation prepared in METU Computer Engineering Department for the course CENG492 Spring 2010

5. http://en.wikipedia.org/wiki/Configuration_management

6. <http://senior.ceng.metu.edu.tr/2011/ballmerpeak/>

1.5 Document Overview

- This document is divided into six (6) sections, the Introduction, the Organization, the Configuration Management Process, Project Schedule, Project Resources and Plan Optimization. The organization of this document is as follows:
- **1. Introduction:** This section presents preliminary information about this document.
- **2. Organization:** This section describes organization and responsibilities of team members and necessary tools to support this structure.
- **3. Process:** This section describes the configuration management process to be used.
- **4. Schedule:** The milestones for CM and the related deadlines are mentioned.
- **5. Resources:** This section describes the resources that Ballmer Peak have.
- **6. Plan Optimization:** This section gives information about plan adjustment in an unpredicted situation.

2. The Organizations CM Framework

2.1 Organization

During the project management process all the members of Ballmer Peak contribute to CM by working actively to develop a successful project. Due to the management constraints, 4 sub teams that have different responsibilities are arranged from CM group. CCT will be the supervisor group of the whole framework.

- **Software Development Team:** Software Development Team is mainly responsible for implementing the modules of Bookwiser project and making the changes that are requested by Testing Team. This Team will also be responsible from the releases.
- **Testing Team:** Testing team will be responsible for two main purposes. First purpose is to check whether the modules meet the requirements. Second purpose is to give feedback about the modules and want to change requests if necessary.
- **Configuration Control Team:** Configuration Control Team will supervise all the activities of other groups. However, the main responsibilities of this group are review SCRs, accept or reject SCRs and monitor SCRs.

- **Configuration Management Team:** Configuration Management Team is responsible for maintenance of the CM organization. Hence, this group will keep the Configuration Management Plan up-to-date.

2.2 Responsibilities

Each CM Team and their responsibilities are explained in part 2.1. In order to increase the functionality, Bookwiser project has been divided into several parts. CCT is responsible for all the system inconsistencies and system faults. Moreover, all team members have the following responsibilities;

- Conforming to CM schedule
- Commenting about changes before committing resources through SVN
- Emailing people about SCR

2.3 Tools and Infrastructure

Platforms that are planned to use are as follows:

2.3.1 Software Development:

Geany: “Geany is a text editor using the GTK2 toolkit with basic features of an integrated development environment.”¹

2.3.2 Version Control:

SVN: “Subversion (SVN) is a version control system initiated in 2000 by CollabNet Inc. It is used 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.”²

All team members will update the source code on METU CEng “BallmerPeak” account. By using SVN, all the works that is done by group members can be changed and accessed simultaneously by other group members.

2.3.3 Project Management

Trac: “Trac is an open source, web-based project management and bug-tracking”³ tool.

We will use Trac to monitor our development process. It will help us track our work according to the milestones defined in our schedule on the website.

2.3.4 Libraries

OpenCV: “OpenCV (**O**pen **S**ource **C**omputer **V**ision) is a library of programming

functions for real time computer vision.”⁴

We are going to implement an “Augmented Reality” based project. Since the real-time computer vision techniques will be the most important part of the project, we will use OpenCV library to handle those operations.

¹ <http://www.geany.org/>

² [http://en.wikipedia.org/wiki/Subversion_\(software\)](http://en.wikipedia.org/wiki/Subversion_(software))

³ <http://en.wikipedia.org/wiki/Bugtracker>

⁴ <http://opencv.willowgarage.com/wiki>

3 Configuration Management Process

3.1-Identification

Configuration Items in our project can be categorized into three group.

3.1.1.Source Code

Source code items are the most important configuration items in the project. All members of the Ballmer Peak are responsible for management of the source code. According to the their functions, source code is classified into three sub-item:

Object recognizer component: processes the video frames and identify the objects in the library scene.

World model component: controls the inner and outer behavior of the Bookwiser

Graphical user interface component: provides user functions and displays the processed video frame.

3.1.2.Data

Data of the Bookwiser are managed by databases. There are two database of the system. Library database stores and provides textual information of books and shelves. Object recognition database stores and provides information for the recognition of the objects(i.e. SURF features).

3.1.3.Documentation

Documentation is also a part of the CM process. The documents are listed below:

Reports:

- Project Proposal
- Requirements Analysis Report
- Initial Design Report
- Detailed Design Report
- Configuration Management Plan

Web Documents:

- Web page
- Web blog

Development Process:

- Weekly Reports
- Facebook Entries

Users Manuals

3.2-Configuration Management and Control

3.2.1-System Change Requests

Minor and Major changes on CI's are handled differently. Minor changes are automatically done by SVN. Major system changes are done by creating tickets on Trac system. A such request consists of:

- Member Name
- Date
- Description
- Related Items
- Due Date

3.2.2-System Change Evaluation

In addition to the Trac, system change requests are discussed in team meetings. After all the opinions about a request are evaluated, A member is assigned to the request is rejected.

3.2.3-System Change Implementation

If a system change request is approved after evaluation process, the assigned member creates a ticket and govern the implementation process. All the configuration items which the change affects are determined. The implementation is done over these items.

3.3-Configuration Status Accounting

CSA is recording and reporting of information needed to manage configuration items effectively. This is crucial for all members and other related individuals to keep track of the CM process. Therefore, when a source file in SVN is updated, the updater should also inform all the other members and related individuals about the details of the update. This provides the development to be followed easily. Living Schedule is updated as well in regards of the accounting mechanism.

3.4 Auditing

To develop a successful project, managing the configuration audits is a very critical

issue. Throughout the project, auditing is being done at regular time intervals. With this process we are aiming to see the possible defects before they occur. We decided to make main audits before each milestone and at the same time we will continue to discuss our problems in our regular meetings. The discussions will be mainly about the design and implementation, the work done during that week, change requests and status of the configuration items. During the discussions, all the members of the group must tell their own opinion about the problem to collect different solutions from different aspects. The results of these audits will be stored in our SVN repository.

4 Project Schedules and CM Milestones

The living schedule for the Bookwiser project is planned due to needs of the customer and the project development dynamics . The project is divided into components that makes developers separately finish the assigned tasks and related schedule about the tasks are listed in the “Living Schedule” part.

The living schedule of the project and the list of project milestones are provided under The Bookwiser Project Web Site [6] .

The important project milestones are as follows:

Project Milestones:

Mar 29, 2011 First Development Snapshot, Demo
Apr 28, 2011 Tests for the First Release
May 10, 2011 First Release, Demo
May 11, 2011 Tests for the Final Release
May 23, 2011 Development of Final Package, User Manual

5 Project Resources

For all the Configuration Management purposes the tools that mentioned below will be used:

Apache Subversion (SVN) is a software versioning and a revision control system . Developers should use subversion to maintain current and historical versions of files such as source code, web pages, and documentation. During the whole project development stages, the development steps will be followed using this tool.

Trac: Trac is an open source, web-based project management and bug-tracking tool that allows hyperlinking information between a bug database, revision control and wiki content. It also serves as a web interface to the SVN revision control system. This tool will be used for bug tracking purposes.

Netbeans: Netbeans is an integrated development environment (IDE) for developing with Java, JavaScript, PHP, Python, Ruby, Groovy, C, C++, and others. This tool will be used all development stages of the Bookwiser project, allowing easy database management and C++ code development / compilation using OpenCV libraries.

Adobe Dreamweaver: Adobe Dreamweaver is a web development application. For the building and update phases of Bookwiser web page, this tool will be used.

Bookwiser Project Website: Our project website , Bookwiser Website [6], is an important resource that can be used by developers to get information about all the planning , scheduling and development phases of the Bookwiser project.

6 Plan Optimization

Coordination of all the planning, scheduling and development phases of the Bookwiser project will be done using the Configuration Management Plan and all this updates will be controlled. All the bug observations will be reported and be followed using the Trac platform and the version controlling of the source code will be done via SVN tools.

The current situation of the living schedule and the development process will be taken into the consideration each week on weekly meetings of team members. Living schedule can be updated at these meetings to catch up to the milestone deadlines for the project. Critical parts of the Bookwiser project can be moved to the earlier dates of the living schedule if it is decided by all the team members on the weekly meetings.

Online meetings can be used by team members to optimize the schedule of the project in emerging situations.

Besides that all the team members will stick to the living schedule and the schedule will strictly be followed during all the development process.