26.3.2010

CONFIGURATION MANAGEMENT DOCUMENT

OCEAN’S 4

SEHIR COBANI

BURAK TIKNAZ 1502749
FATİH PEHLİVAN 1502590
OKAN ÖZMEN 1502947
YAKUP TURGUT 1502780

Table of Contents

1 INTRODUCTION

1.1 PURPOSE OF CMP

The purpose of this document is to maintain the continuity and integrity of our project "Şehir Çobani" during the development process. In all comprehensive projects, the project is separated into several parts and modules. Generally, these parts are implemented by different people so that the project can be developed concurrently. However, working with many people in parallel requires working in a managed and planned way. Because of this reason, it can be said that all projects require configuration management plan. Thanks to configuration management , the changes made by one person can be easily monitored to all people so that waste of time which occurs because of person's being unaware from the updates and changes made in the system is prevented.

1.2 SCOPE OF CMP REPORT

This document is prepared by Ocean's 4 members for the project named "Şehir Çobani" to manage the configuration and to handle changes made to software, test and documentation throughout the development process of our project.

This document gives information about the responsibilities of our group members and the way we organize our team. In addition it defines CM processes, CM milestones and the schedule of the development process.

1.3 DEFINITIONS ACRONYMS AND ABBREVIATIONS

- CMP: Configuration Management Plan
- CM: Configuration Management
- GUI: Graphical User Interfaces
- SCR: Specification Change Request
- SVN: Subversion
- CI: Configuration Item
- Metu: Middle East Technical University
- Ceng: Computer Engineering
- SDT: Software Development Team
- STT: Software Testing Team
- CCT: Configuration Control Team
- CMT: Configuration Management Team

1.4 DOCUMENT REFERENCES

- Software Engineering: A Practitioner's Approach – Roger S. Pressman, 6th Edition
- Software Configuration Management Plan, Presentation prepared by METU Computer Engineering Department for the course Ceng 492
- Our Requirement Analysis Report
- Our Final Design Report
- http://standards.ieee.org/reading/ieee/std_public/description/se/1042-1987_desc.html
- http://netbeans.org/
- http://trac.edgewall.org/
- http://www.wikipedia.org/
- http://subversion.tigris.org/
- https://senior.ceng.metu.edu.tr/trac/oceans4

1.5 DOCUMENT OVERVIEW

This document consists of mainly six parts.First part is the introduction. In this part, the purpose of this document, its scope , also the definitions acronyms and abbreviations for the clarification of the document are explained . At the end of introduction part, the sources we referred are stated.

Second part is the organization. In this part, the organization and responsibilities of all team members for CM are stated. Moreover, the infrastructures and the tools we will use in configuration management are explained in detail.

Third part is the CM process. In this part, the identification, management, and the auditing of the CIs are stated.

Fourth part is project schedule. In this part, the milestones specified earlier and general schedule of our project take place.

Fifth part is project resource. In this part, the project resources which will be necessary for CM is enucleated.

The last part is plan optimization. In this part, the methods and strategies we will use in the change of milestones or in some unexpected situations are clarified.

2 THE ORGANIZATION OF CM FRAMEWORK

2.1 ORGANIZATION

Our group Ocean's 4 consists of 4 members:

- Burak Tıknaz

- Fatih Pehlivan
- Okan Özmen
- Yakup Turgut

It can be said that our group is organized in four inner teams in itself. Thus, each member will work actively in all teams. However in these teams, the workload of our group members may change differently.

2.1.1 Software Development Team

Software Development Team is mainly responsible for implementing the modules of Şehir Çobanı project and evaluating the SCRs coming from STT and implementing these requests.

2.1.2 Software Testing Team

Testing group is responsible for evaluating whether the modules implemented by SDT meets the requirement or not. In addition, after testing, based on test results, testing group may make suggestions on implementations to improve the system and may fix the necessary situations.

2.1.3 Configuration Control Team

Configuration Control Team will be the group which controls the activities of other groups. This team will also review SCRs, evaluates them and accepts if it necessary or rejects if it is not proper.

2.1.4 Configuration Management Team

CMT will update the CM according to the current status and development process of the project.

2.2 RESPONSIBILITIES

We have mentioned responsibilities a little in the organization part. Additionally, we can say that, despite all of us take part in all kind of implementation. In some implementation part, some people may take bigger role.

Burak will be mostly dealing with accident&violation message class implementation and following distance&overtaking&crossroad message class implementation. In addition, Zigbee implementations are another part in which Burak takes place.

Yakup will be mostly dealing with direction&calculateStatistics class implementation and traffic density message class implementation. Moreover, simulation engine implementation is another part of our project in which Yakup takes big role.

Fatih and Okan will be mostly dealing with GUI Design, GetGpsData class implementation, map class implementation, Emergency Services class implementation and documentation.

As a final responsibility of all group members, every change should be reflected as soon as possible in a well documented manner for the sake of maintenance and understandability of the project.

2.3 TOOLS AND INFRASTUCTURE

Platforms that we are currently planning to use are as follows:

2.3.1 NetBeans

The NetBeans Platform allows applications to be developed from a set of modular software components called modules.

The NetBeans project consists of an open-source IDE and an application platform that enable developers to rapidly create web, enterprise, desktop, and mobile applications using the Java platform, as well as JavaFX, PHP, JavaScript and Ajax, Ruby and Ruby on Rails, Groovy and Grails, and C/C++.

2.3.2 Trac

Trac is an enhanced wiki and issue tracking system for software development projects. Trac uses a minimalistic approach to web-based software project management.

It provides an interface to Subversion (or other version control systems), an integrated Wiki and convenient reporting facilities.

Trac allows wiki markup in issue descriptions and commiting messages, creating links and seamless references between bugs, tasks, change sets, files and wiki pages. A timeline shows all current and past project events in order, making the acquisition of an overview of the project and tracking progress very easy. The roadmap shows the road ahead, listing the upcoming milestones.

2.3.3 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 (CVS).

All team members will properly update the source code on Metu Ceng Ocean's 4 account. Since four members may work on the project simultaneously and each member should understand the change of other members, SVN use will be essential.

3 THE CM PROCESS

3.1 IDENTIFICATION

3.1.1 Source Code

Our project's source code resides on SVN repository which our department provides us. By using this repository, we, all group members, can check in and check out SVN repository and synchronize our codes with no problems. The source code of Sehir Cobani project is mainly written in java programming language.GPS and some other technologies can also be used in the code. The source code of the project is developed using NetBeans.

Sehir Cobani's source code can be divided into two main parts. First one is a GPS software for a mobile GPS device. The other part is a desktop application for the Emergency Services.

3.1.2 Database

Almost all accident data and road statistics will be kept in a consistent database. We are planning to use MySQL database. The kept information will be reachable from desktop application also.

3.1.3 Additional Data

The additional data which is not stored in the database will be stored in a separate directory in the SVN file hierarchy. This data includes images, style files, configuration files, and so on.

3.1.4 Documentation

The documentation includes

- Installation Manual
- User Manual
- Developer API

3.2 MANAGEMENT AND CONTROL

We're going to use SubVersion revision control system and Trac project management system in order to manage and control configuration management process. Both services are provided by METU Computer Engineering Department and can be accessible from Ocean's 4 Trac.

Configuration Management and Control consist of various tools and procedures for various steps of software development process:

3.2.1 Engineering

This process involves inspecting the suitability of the software for its intended use and it includes

- Verifying the consistency between the design and the implementation
- Proposing a roadmap to compensate for any incapability.

3.2.2 Development

In order to maintain the development management and control, we use

- Sun Microsystems Code Conventions to facilitate a standard which avoids miscommunication and forms a more extensible product.
- JavaDoc to generate documentation
- Naming conventions for module, class and method names so that confusion is avoided and necessary modules can be located easily.

3.2.3 Change Requests

The primary medium to make new requests about the project is Trac system. This system can be used to add tickets and modify milestone requirements. All tasks like “enhancement requests”, “new feature requests” and “defect tracking” can be managed and controlled by ticket system.

The ticket database is capable of keeping track of this information:

- Type of the request: task, enhancement or defect
- Summary of the request
- Detailed information about ticket
- Related component
- Date / time of the request

These requests can be made by any member of the group and then evaluated by every member.

3.2.4 Defect Tracking

All group members are responsible for testing and reporting the defects in the software. Then the team will keep track of those defects and facilitate the proper notification to the member who is responsible for development of that package class etc. These defects include

- The name of member reporting the defect or defects
- The indication of defect or defects
- The revision number of modules on which the defect or defects observed
- The priority of the defect or defects
- A recipe (if possible) for the reproduction of the observation of the defect or defects

3.3 AUDITING

Auditing involves evaluating the project in order to ascertain the validity and reliability of the system and taking actions according to evaluation results. It will be performed after the last release.

There are three types of auditing as follows.

- Functional Audits: This will be performed to test the software to see whether it performs in accordance with requirements in the baseline.
- Physical Audits: This will be performed to assure that the software contains all of the required components, documents and data.
- Process Audits: This will be performed to compare and contrast the manner in which the end product is produced to the written procedures.

4 PROJECT SCHEDULE

We have already prepared living schedule which has tasks and milestones. Since we designed the components separately in the first term, it will be easy workload sharing. The schedule of our project can be accessed through our project website [1].

We will first build a GUI, map and message class which is the heart of the Sehir Cobani. Then, we are planning to continue with development of Accident and Following distance classes. After that we will develop other parts of the Sehir Cobani.

After second snapshot, we hope, all major components will be implanted. Then, all work is about testing, bug fixing and maybe adding some more capabilities to the Sehir Cobani. At the end of the semester, we are planning to write documentation and manuals.

4.1 Milestones

These are the intended milestones for Sehir Cobani:

- First Snapshot – 6 April 2010
- Second Snapshot – 13 May 2010
- Final Package – 3 June 2010
- Documentation – 10 June 2010

5 PROJECT RESOURCE

The following items will be used as project resources

- SVN : Revision Control System
- NetBeans IDE : Development Environment
- Trac: Issue Tracking System
- Web Site: Project Development News

6 PLAN OPTIMIZATION

Although we have written our configuration management plan here, there may be small changes on the dates and orders of the process to be done since it is done on predictions. However, we will try to stick to configuration management plan. If some changes are needed in the schedule, we are going to arrange meetings with our group members and update the schedule accordingly and while updating the schedule, the milestones will be the most crucial factor.

[1] http://senior.ceng.metu.edu.tr/2010/oceans4/