

CENG 492

Computer Engineering

Design 2



Configuration Management

Plan Report



Geeks In Action

M. Oğuz ŞEN	:	1395532
Talat ÖZER	:	1347806
Nur Muhammet ARINÇ	:	1448364
Cuma KILINÇ	:	1395193
İshak YAPAR	:	1348101

Contents

1 Introduction	3
1.1 Purpose of the Document	3
1.2 Scope of the Document	3
1.3 Definitions, Acronyms and Abbreviations	4
1.4 Document References	4
2 The Organizations CM Framework	4
2.1 Organization	4
2.2 Responsibilities	5
2.3 Tools and Infrastructure	6
2.3.1 Project Management Tools	6
2.3.2 Technical Tools	6
3 The CM Process	7
3.1 Configuration Identification	7
3.1.1 Game Screen Graphics	7
3.1.2 User Interface	7
3.1.3 Network	7
3.1.4 Game Data	7
3.1.5 Artificial Intelligence	8
3.1.6 Physics	8
3.1.7 Audio	8
3.1.8 Documentation	8
3.2 Configuration Management and Control	9
3.2.1 System Change Request	9
3.2.3 System Change Evaluation	9
3.2.4 System Change Approval	9
3.2.5 Implementation of Approved Changes	9
3.3 Configuration Status Accounting	10
3.4 Configuration Auditing	10
4 Project Schedules	10
5 Resources	11
6 Plan Optimization	11

1 Introduction

This document is the configuration management plan of Football Game for Linux project, performed by the group Geeks In Action.

1.1 Purpose of the Document

Football for Linux project cannot be classified only as a trivial software project. Input from several distinct disciplines is required such as (network, graphics, AI, physics etc.) and five senior computer engineering students are working on it currently. However, the project can be extended in a more complicated manner. The design document of this project is not as detailed as one may like and hence, possible missed parts have to be figured out in implementation phase. It is certain that some design elements simply may become useless and deficient or more useful methods could be discovered. **Because** of all these factors, the product should be revised constantly by all group members. Configuration of the software includes all the components, relations between the components, all the structures and relations forming the development process. The formal procedures and protocols put in place to revise and refine the configuration are collectively called the configuration management plan.

1.2 Scope of the Document

This document presents the configuration management structures and activities of Geeks in Action. The general information about this report is provided in the “Introduction” chapter. Next; in the management chapter, an overview of group structure is presented. Third chapter explains the configuration management process of Geeks in Action and how it is maintained. Scheduling and resources are presented in fourth and fifth chapters respectively.

1.3 Definitions, Acronyms and Abbreviations

SCM	Software Configuration Management
CI	Configuration Item
SCMP	Software Configuration Management Plan
CMP	Configuration Management Plan
CM	Configuration Management

1.4 Document References

Standards and guidelines used in the preparation of this document are from the following references:

- “Software Configuration Management”, The presentation prepared in METU Computer Engineering Department for the course CENG492
- Pressman, Roger S. Software Engineering: A Practitioner's Approach, Sixth edition. New York, NY: McGraw-Hill
- IEEE Standard for Software Configuration Management Plans (IEEE Std 828-1998)

2 The Organizations CM Framework

2.1 Organization

Since every member has nearly same experience about design patterns and concepts of this project, decisions about the project are made by group consensus. This situation yields our team to have democratic decentralized structure. Communications within the group is horizontal. Like many other software engineering projects, presence of a team leader is vital in game projects. We cannot evade this fact, so we chose a team leader for coordination and interactions within our team. While making decisions, our team leader consults each team member in order not to go against our team structure.

Assignment of roles in a team to each team member is listed as below:

Team Leader and Initiator : M. Oğuz Şen
Initiator and Optimist : Cuma Kılınç
Devil's Advocate and Time Keeper : Talat Özer
Recorder : Nur Muhammet Arınç
Gate Keeper : İshak Yapar

2.2 Responsibilities

As a result of having different modules and tasks in project, division of labor is required. It can be seen in this table.

	Talat	M. Oğuz	Cuma	Nur Muhammet	İshak
Game Graphics	X	X			
Modelling and Motion Capture	X	X			
Network			X		X
Audio				X	
Ball Physics	X				
AI		X			
User Interface			X		X
Database			X	X	

2.3 Tools and Infrastructure

2.3.1 Project Management Tools

- **SVN:** SVN (SubVersion) is a version control system. Since project is comprised of different modules, SVN is used to track every member's modifications and improvements.
- **Trac:** Trac is a project management, communication and bug tracking tool. Every module, task bug etc. can be assigned to group members to separate everyone's responsibilities. Moreover it is a tool that enables communication and shows project progress.
- **Web Page:** All documents and project progress can be seen via web page.

2.3.2 Technical Tools

- Our project runs on Linux operating system. The reason is that Linux is open source and team members are supporters of open source and free software.
- We are developing our game on C++.
- In terms of graphics, we are using C++ for implementation and Irrlicht as graphics library and game engine.
- In terms of graphics, OpenGL is used for graphics rendering. The reason is that all of team members are experienced about C++ programming language and OpenGL.
- We are going to use FMOD as a sound library. Since FMOD is a very powerful library and it is free to use it in non commercial projects; we chose to use it in our project.
- For creating and editing images for texture, we are going to use GIMP, since it works also with Linux environment.

- For environments (stadiums etc) and players, we are using free models. However, we are going to add animations to player models using Motion Capture system.

3 The CM Process

3.1 Configuration Identification

For configuration identification of the software engineering projects, attributes that define every aspect of configuration items are identified. Configuration items of this project consist of 8 main parts.

3.1.1 Game Screen Graphics

- **Static Models:** Stadiums, astro pitches, indoor saloons, streets and various suburb areas.
- **Rigged Models:** Footballers, referees, balls
- Skydome
- Terrain
- **Weather Effects:** Sunny, rainy, snowy weather conditions; day and night modes

3.1.2 User Interface

- **Main Menu:** Mode Selection, Settings, Multiplayer options.
- **In Game Menu:** Match status, tactic and lineup options, substitution options

3.1.3 Network

- Local area network connection with UDP packets

3.1.4 Game Data

- Ball position
- Footballer and referee locations on the playground.
- Match score
- Ball possession status: Freely moving, possessed by a footballer

3.1.5 Artificial Intelligence

- Tactics: Offensive, defensive playing
- Goalkeeper actions
- Heuristic methods for footballer control

3.1.6 Physics

- Collision detection
- Projectile motion: Simulation of ball motion in 3D space
- Footballer speed

3.1.7 Audio

- Sound tracks: Music for main menu screen. (also for sub-menus)
- Sound effects: Ball sound, referee whistle, spectator sounds
- Narration: Commentator sounds

3.1.8 Documentation

- Design reports of this project
 - Project Proposal
 - Requirement Analysis Report
 - Initial Design Report
 - Detailed Design Report
 - Configuration Management Plan Report
 - Test Specifications Report
- Development Process Reports
 - Weekly Reports
- User Manuals
 - Installation Manual
 - Player's Manual
 - Development Guide

3.2 Configuration Management and Control

The process for submitting, evaluating and implementing configuration change requests can be analyzed in four stages.

3.2.1 System Change Request

Any type of change requests can be performed by everyone in our group. Minor requests are going to be directly put in to the project, since they are going to be handled by SVN. When the need for a major change arises, the process is going to be managed by Trac system. Each request is going to be represented as a ticket and this ticket is going to be opened to general access and every member of the group is going to be informed of the change request.

3.2.3 System Change Evaluation

The change request tickets in Trac are going to be the main stage for debates on the issue. Furthermore, group members are going to be able to talk about change requests in weekly and extra meeting times.

3.2.4 System Change Approval

Each team member is responsible about his specific labor. In case of vagueness on any kind of change request, the outcome of the issue is going to be determined by group consensus.

3.2.5 Implementation of Approved Changes

After the approval of a change request, the implementation part of that request is assigned to the group member that has the responsibility of the corresponding module(s). After the implementation is finished, the member uploads the code to the SVN and every group member is informed about that change. Since only five people are currently working on the project, all implementations are going to be considered by all group members.

3.3 Configuration Status Accounting

Configuration status accounting consists of the performed recording and reporting actions related with CIs. It tells the group, instructors and the assistant about the current status of the project. In case of changes in the configuration, our assistant is going to be informed by Trac system prior the weekly meetings. Furthermore SVN is going to keep the track of all the changes related with our project. When a group member makes changes on the source code, he is going to write comments on SVN and Trac. Description and the purpose of the change are going to be explained clearly in the comments. All the members are going to take care for a possible corruption or loss of data on the project. Lastly; the changes are going to be enumerated with date of the change, and they are going to be placed in a text file to create a change log for correct evaluation of the project progress.

3.4 Configuration Auditing

For this project, our team leader has the responsibility for maintenance of the reliable configuration version. Important changes are notified to team members by team leader via meetings or other suitable channels such as Internet, phone call etc. The group reviews the state of the project and makes required modifications to the current configuration.

4 Project Schedules

The main milestones of our project are listed below. You can also find the full living schedule.

- **First Development Snapshot Demo:** Modular version of the first semester's prototype is going to be exhibited. We are going to identify and add new modules upon the first semester's prototype. Its deadline is 06.10.2010.

- **Pre-First Release Prototype:** It is planned to finish the skeleton of the game. Project modules are going to be completed by this prototype.
- **First Release:** It is an official milestone in order to finish the project. We are not going to make any architectural change in the project after this milestone.
- **Final Release:** It is the end of the project. We are going to finish all modules till then, and we are going to complete integration, testing and documentation.

5 Resources

The following tools are used for CM activities by our group, Geeks in Action:

- SVN
- Web Site
- Mail Group
- Trac

Every member of the group has to learn using Trac. Each member is going to use Trac in order to show their progress. These CM activities are going to help to make our efforts more efficient. And a steady software development cycle will be achieved. Weekly report, weekly backups, situation reports will be included.

6 Plan Optimization

We divided our CM responsibilities between our five members according to their interests. Each member is responsible for himself. We mostly communicate via phone call. In addition, we are making regular weekly meetings for more control over the group and also we have weekly meetings with our assistant.

Moreover, we control our process with small demos for consistency of our project.