Software Engineering Group 2 (2009-2010)
Software Project Management Plan

Nick De Cooman

October 21, 2009

Contents

Chapter 1

Introduction

1.1 Project overview

The goal of this project is to design and implement the software behind an online auction website such as eBay. The codename of our project is *S*alesmen.

This project is embedded in the Software Engineering course, which is designed for senior students of Computer Science and Engineering at the Vrije Universiteit Brussel. It has a threefold purpose:

1. Get familiar with the software development process and learn to work in a team.

2. Develop a system that supports the functionality of an auction website. This means that a user will be able to sell or buy items in an online auction.

3. Develop and maintain a general website on our project. Up to date versions of all documents such as minutes, timesheets and deliverables will be available at any time. This site will be hosted at `http://wilma.vub.ac.be/~se2_0910/`.

The software of the application will be written in Java and distributed under an Open Source license.

1.2 Definitions and acronyms

SCMP	Software Configuration Management Plan
SDD	Software Design Plan
SPMP	Software Project Management Plan
SQAP	Software Quality Assurance Plan
SRS	Software Requirements Specification
STD	Software Test Document

1.3 Project deliverables

Deliverable	Available on
Minutes	3 days after meeting
Timesheet	Every Monday before 12h00
SPMP	Draft will be ready on October 26, 2009
SCMP	Draft will be ready on November 2, 2009
SRS	Draft will be ready on November 2, 2009
SQAP	Draft will be ready on November 23, 2009
SDD	Draft will be ready on November 16, 2009
STD	Draft will be ready on November 30, 2009

1.4 Reference materials

- `http://tinf2.vub.ac.be/~dvermeir/courses/software_engineering/slides.pdf`
- Eric J. Braude, *Software Engineering - An Object-Oriented Perspective*, John Wiley & Sons, 2001. ISBN 0-471-32208-3.

1.5 Revision history

Version	Date	Name	Description
1	21/10/09	Nick De Cooman	Initial draft

Chapter 2

Project Organization

2.1 Process model

The chosen proces model will be the spiral model, with 2 iterations. Unlike the waterfall process, we will be able to show partial versions to the customer for feedback, reducing risks such as faulty requirements.

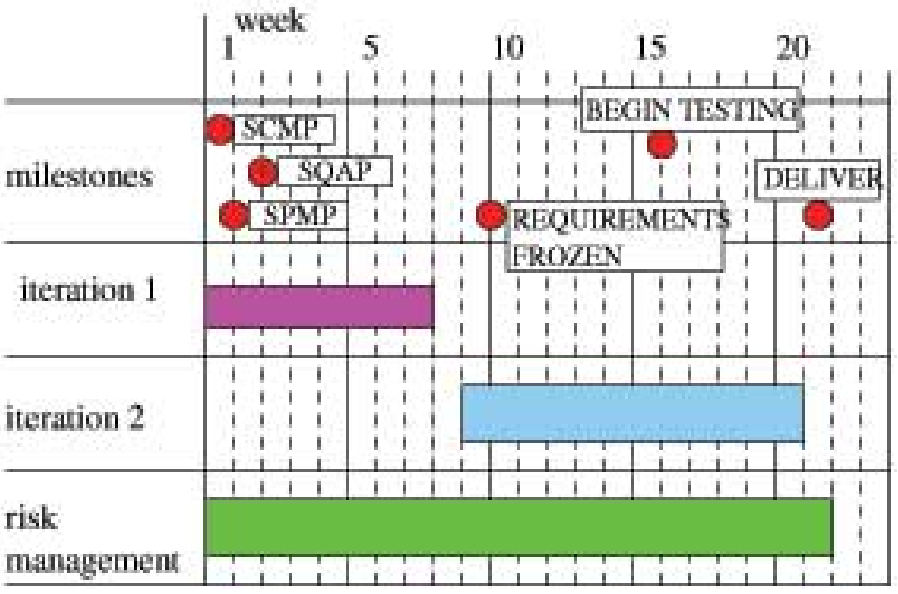

Figure 2.1: A general example of the spiral model with 2 iterations.

In our case, this will be defined as:

Workflow	Completion date	Deliverable to customer
Flow 1		
Requirements	09/11/09	??/11/09
Design	23/11/09	
Implementation	07/12/09	
Testing	16/12/09	2*/12/09

Flow 2
Will be defined later.

2.2 Organizational structure

The structure of our team will be organised as followed:

2.2.1 Project Manager

- Head of the project team
- Deliver the SPMP
- Risk Analysis
- Keep track of the project's progress
- Manage timesheets
- Prepare and chair meetings
- Spokesman to organizational boundaries and interfaces
- Motivate and improve internal collaboration

2.2.2 Project Secretary

- Resume meetings
- Deliver minutes

2.2.3 Configuration Manager

- Deliver the SCMP
- Manage revision control system
- Install and maintain used tools
- Make backups of all project documents

2.2.4 Quality Assurance Manager

- Head of quality team
- Deliver SQAP and STD
- Design and implement test framework
- Coordinate testing as defined in STD
- Verify the general quality of the product
- Verify the quality of documents

2.2.5 Requirements Manager

- Deliver SRS
- Check whether the requirements are implemented
- Look for extra functional requirements
- Interview customer related to the requirements
- Present final requirements to the customer

2.2.6 Design Manager

- Head of design team
- Deliver SDD based on the SRS
- Manage design of the system
- Check whether implementation is based on the SDD
- Design and manage database according to SRS and SQAP
- Present design to the customer

2.2.7 Implementation Manager

- Head of implementation management
- Manage the integration of different parts of the code
- Track errors
- Define implementation proces based on the SDD
- Manage documentation of code

2.2.8 Webmaster

- Develop and maintain project website
- Upload documents, after approval of the project manager

2.3 Organizational boundaries and interfaces

The external communication with the customer will only happen via the project manager, the requirements manager and the design manager. The project manager reports major problems, progress status and other organisatorial issues. The requirements manager discusses the functionality of the system and the design manager will present the design of the system to the customer.

2.4 Project responsibilities

The responsibilities, as defined in 2.2, will be associated to the following persons:

Role	Effective	Assistant
Project Manager	Nick	Patrick
Project Secretary	Jonathan	Wouter
Configuration Manager	Jorne	Sina
Quality Assurance Manager	Patrick	Bart
Requirements Manager	Wouter	Jonathan
Design Manager	Bart	Nick
Implementation Manager	Sina	Jorne
Webmaster	Sina	Wouter

However, this does not mean that a person is only responsible for his/her own function. Every team member can be asked for assistance in all processes in function to achieve the end goal.

A person responsible for creating a deliverable also needs to be aware of the fact that

- Documents will be writen in LaTeX-format, using the provided template.
- Documents need to be ready in time (as defined in 1.3), and sent to the project manager for approval.
- Documents needs to be up-to-date during the entire project.

Chapter 3

Managerial process

3.1 Objectives and priorities

During the development of the system, the following objectives should be kept in mind:

1. The project should meet the requirements of the customer
2. Deadlines have to be respected
3. Reusability and extensibility of the software is very important
4. The system should be stable

3.2 Assumptions, dependencies and constraints

- No assumptions were made.
- Dependencies
  For the project to succeed, it will depend on the knowledge and the motivation of the team members
- Constraints:
  - Only open-source software is to be used
  - The product should run in a Linux environment
  - The product should run on the Wilma server

3.3 Risk Management

Developing software introduces a lot of risks. Every meeting these risks will be discussed in order to reduce or eliminate them.

3.3.1 Google goes down for a certain period

Criticality: ++
We need to make backups of our files and documentation. This will be performed by the configuration team.

3.3.2 Someone doesn't spend enough time on his job

Criticality: ++++
At first this will be discussed at the next meeting and a suggested action will be given. Thereafter a warning will be given. Possibly there will be further actions. If its absolutely necessary, a member can be dismissed.

3.3.3 Deadline will not be achieved

Criticality: ++++
Not achieving a deadline, can bring the timeschedule in danger. Because of this a team member should report this to the project manager who can reorganise the tasks, as soon as he/she perceives a delay.

3.3.4 Somebody becomes ill

Criticality: ++++
This risk has already been considered in the beginning of this project. This is why there is an Assistant(formerly called Backup) for every Management position. If a leader or manager gets ill, the assistant of that function should be able to fully understand his function and replace that leader or manager, for a certain period of time.

3.3.5 The client can't make an appointment before the deadline

Criticality: +
The client should be contacted as soon as possible. *Update: The client has been contacted on Friday, October 23, 2009 and an appointment has been made on Monday, October 26, 2009.*

3.3.6 The wrong questions have been asked to the client

Criticality: +++
There should be at least two appointments with the client before the first version of the SRS becomes available.

3.3.7 The requirements are misunderstood or misinterpreted

Criticality: +++
The interview with the client will be recorded and will be made available to all team members. This interview will be relistened by the Quality Assurance manager to reduce misinterpretations. In order to avoid ambiguity, there will be a presentation of the requirements during another meeting with the client.

3.4 Monitoring and controlling mechanisms

3.4.1 Meetings

Meetings will be held every week. The topics to handle at the meeting will be defined in an agenda and will be sent to every team member before the start of the meeting. Minutes will summarize the meetings and will be available on the project website within 3 days after the meeting.

Should one not be able to attend the meeting, he/she is requested to inform the project manager within 3 hours before the start of the meeting.

3.4.2 Timesheets

A global timesheet will be available every Monday before 12H00. Team members will need to submit their timesheet on Sunday before 23h59 for approval by the team manager.

Chapter 4

Technical Process

4.1 Methods, tools and techniques

- Google Code
  There has been chosen for Google Code because it has some handy features:
  - Revision control system: Subversion in our case, see below.
  - Issue tracker
  - Wiki (is sometimes handy e.g. for meeting agenda).
  - File download server (for our software and document releases).
- Subversion
  This is a centralized version control system chosen for the following reasons:
  - It is easy to refactor the source code structure, while preserving files' history.
  - The whole repository has a single revision that is incremented after each commit.
- Eclipse
  Eclipse is a software development environment comprising an IDE and a plug-in system to extend it. It is used to develop applications in Java. It was decided to work with Eclipse because it has a very userfriendly interface and because of its popularity. Also it's known by most of the team members.

- LaTeX
  LaTeX has been chosen to document this project because it's internationally known and commonly used. Also, a few members were interested in learning this language.

More specific tools will be discussed later.

4.2 Software documentation

See the SQAP.

4.3 Project support functions

Throughout the entire project, Joeri De Koster and Dirk Vermeir will be available for any help. For Wilma related problems, Dirk Van Deun can be contacted.

Chapter 5

Work packages, schedule and budget

5.1 Work packages

All costs are expressed in man-hours.

Package	Deliverable	Total estimated time
Organizational	Meetings (+/- 30)	210 Hr
	Timesheets	25 Hr
	SPMP	30 Hr
	SCMP	20 Hr
Analysis	SRS	40 Hr
Design	SDD	50 Hr
Implementation	Source code	250 Hr
Quality Assurance	SQAP	100 Hr
	STD	150 Hr

Please note that the numbers above are just an estimation and will become more precize after every stage in the project.

5.2 Dependencies

At this time, it is only possible to provide 1 general dependency :

- Iteration 2 depends on the completion of iteration 1.