
Edition: 1

Page number: 20

Date: 2-Oct-10

DEVELOPMENT METHODS

2010 PROJECT WORKFLOW

AUTHOR	PROJECT LEADER	QUALITY MANAGER
Name : J de Montcheuil	Name : J de Montcheuil	Name : Andréas Topp
Signature :	Signature :	Signature :

SUMMARY

1 - Introduction.....	3
1.1 Purpose	3
1.2 Document structure	3
1.3 Terminology and abbreviations.....	3
2 - Project objectives	4
3 - Development method principles	4
3.1 Method principles.....	4
3.2 Sprint or step sequence.....	5
3.3 Documentation	5
4 - Business plan steps	6
5 - Project scheduling	6
5.1 Project calendar	6
5.2 Organization and constraints	7
5.3 Supplies delivery and mails.....	7
5.4 Additional information	7
6 - Work to be done	8
6.1 Team organization and kick off	8
6.2 Business Plan redaction.....	8
6.3 Product definition	9
6.4 Concept validation review	10
6.5 Software development.....	10
6.6 End semester reviews	11
6.7 Debriefing.....	11
7 - Quality requirements.....	11
8 - Project evaluation.....	11
Appendix 1 General planning example	13
Appendix 2 Step planning example	14
Appendix 3 - Project leader.....	15
A2.1 Attributions.....	15
A2.2 Advice	15
A2.3 Requirements.....	15
Appendix 4 Software development supplies description	17
A4.1 Step Preparation Document (SPD).....	17
A4.2 Project Records (PR)	17
A4.3 General Requirements Specification (GRS).....	17
A4.4 Release Definition Document (RDD)	17
A4.5 Software Quality Plan (SQP)	18
A4.6 Prototypes	18
A4.7 User Manual (UM)	18
A4.8 Release Acceptance Procedure (RAP)	18
A4.9 Step Test List (STL)	18
A4.10 Code Integrated Documentation (CID)	19
Appendix 5 - Individual activity report model.....	20

1 - Introduction

1.1 Purpose

This document defines schedules and constraints for the development methods project during the 2010-2011 academic year.

All project subjects will refer to this document to define schedule and constraints, and will define project context and statement of work.

1.2 Document structure

This document, after presenting the project's objectives, contains four parts:

- method principles,
- business plan steps
- project scheduling,
- work to be done.

The project scheduling section details project steps and their calendar, and constraints on the project.

The *work to be done* section gives indications on the main activities of the project.

Appendixes:

- planning examples,
- project manager role description,
- supplies description,
- individual activity report model.

1.3 Terminology and abbreviations

Scrum: Agile iterative development method inspired from rugby scrum

Release backlog: In scrum method, the list of features or stories to implement for realizing the release

Sprint: in scrum method a software realization step

Sprint backlog: in Scrum method, the list of features or stories to implement in the sprint with their breakdown in tasks

BP Business Plan

CID Code Integrated Documentation

GRS General Requirements Specification

PDD Product Definition Document

PR Project Records

RAP Release Acceptance Procedure

RDD Release Definition Document

SPD Step Preparation Document

SQP Software Quality Plan

STL Step Test List

UM User Manual

2 - Project objectives

The project is designed to simulate an industrial software development process, applying teachings from:

- development methods,
- quality,
- object-oriented programming,
- event programming and languages (C++ , Java, C#),
- management.

Project will contain two parts:

- development of software,
- business plan writing.

Each subject will define the statement of work for the software to develop. For reasons of work load, full industrial development is not required, but only a usable prototype.

Each subject will also define the project context for business plan work.

Students will be grouped in teams of 6 to 8 students. Subjects will generally be treated by several teams; each team must deal with the subject separately.

During the project, each team will be supervised by a technical manager, on technology and methods, and by a marketing manager on business.

In addition, for some subjects a technical expert will help teams on technical aspects.

3 - Development method principles

3.1 Method principles

Development method will be an iterative method using Scrum agile method principles.

Software development is limited to one release, it will contain:

- a product definition step for project preparation,
- a release definition step,
- an iterative development of the release with three sprints.

The objectives of product definition is to define the project resulting product content, to evaluate project costs and delays, to examine needed resources and benefits of project, and all elements necessary to examine the opportunity to realize the project.

After presenting the results of the product definition step to its teachers, each team will begin the definition of the release to be realized. The objectives are to analyze the requirements for the release and examine the software architecture in order to draft a first version of the release backlog with priorities and preliminary workloads estimations.

Then the release is realized in three sprints, and for each sprint team will select in release backlog new features to implement and will realize a new prototype with all features of precedent prototype and new features. As far as possible, features will be realized in order of priorities so that, if product is not complete at end of project, the most important features are implemented. Last sprint ends by a prototype acceptance with technical manager.

3.2 Sprint or step sequence

Technical manager will play the user or customer role of agile principles. For each sprint we shall have the following sequence:

- the team establishes gatherings of preceding sprint or step : activities realized by each member, quality actions and problems, problems remaining (problems noted during acceptance and other problems not solved), advancement charts, then selects features to implement in the sprint and establishes a detailed planning of the sprint,
- about a week after the begin of the step, these information are furnished to technical manager (SPD - Step Preparation Document) and a scoping meeting is done with technical manager to examine the work to be done during the step,
- the team realizes the planned work and defines the tests to accept the sprint prototype,
- a few days before step end, the team sends to technical manager the Step Test List (STL) and documents specifically required for the step,
- step ends with the presentation of the prototype to technical manager for acceptance (using Step Test List).

3.3 Documentation

Following agile principles documentation will be limited to really useful documents, and documents will be simple, written just in time and integrated to code as far as possible.

So requirements document will be limited to general requirements, without detailed description of function or use cases.

Release definition documentation will be limited to release backlog and some diagrams showing interactions and dependences.

4 - Business plan steps

Business plan is realized in 5 steps, a new version is furnished at end of each step.

First semester version V1 will be realized in 3 steps with a new project of version at each step, with as main contents:

- market study,
- market segmentation,
- target market analysis,
- definition of product and services.

For second semester, version V2 will be realized in 2 steps with a new project of version at each step, with in addition:

- corporate strategy,
- marketing plan,
- finance.

5 - Project scheduling

5.1 Project calendar

There will be 5 steps:

Step		Date
Product definition	Project presentation Kick off meetings (technical and management) Delivery of BP V1-P1 Delivery of PDD V1 Concept validation review	week 4 to 9 oct 10 week 18 to 23 oct 10 10 nov 10 10 nov 10 week 15 to 20 nov 10
Release definition	Delivery of SPD and scoping meeting Delivery of BP V1-P2, Delivery of GRS V1, RDD V0 and SQP V1 Project review Business plan meeting	week 22 to 27 nov 10 9 dec 10 9 dec 10 week 13 to 18 dec 10 week 3 to 8 jan 11
Sprint 1	Delivery of SPD and scoping meeting Delivery of BP V1 Delivery of GRS V2, RDD V1, SQP V2 and STL Semester 1 validation review with P1 presentation	week 3 to 8 jan 11 20 jan 11 20 jan 11 week 24 to 29 jan 11
Sprint 2	Delivery of SPD and scoping meeting Delivery of BP V2-P1 Delivery of RDD V2 and STL Project review with prototype P2 presentation Business Plan meeting	week 31 jan to 5 feb 11 17 feb 11 17 feb 11 week 28 feb to 5 mar 11 week 28 feb to 5 mar 11
Sprint 3	Delivery of SPD and scoping meeting Delivery of BP V2 Delivery of RAP and UM Project acceptance (technical and management) and debriefing Delivery of CID and PR	week 7 to 12 mar 11 1 apr 11 1 apr 11 week 4 to 9 apr 11 9 apr 11

Delivery dates are **latest** delivery dates. Meetings will be planned according to timetable constraints. All team members must attend all meetings.

Calendars proposed by teams will adjust to these dates.

Some lab exercises will be organized during the project to help teams in their work.

At the end of each step, a project review (corrections and presentations) takes place with your technical manager. Moreover at the concept validation review, the semester validation review, and project acceptance, both marketing and technical managers will be present. Additionally, business plan advancement meetings will be scheduled with your marketing manager.

5.2 Organization and constraints

- the teams will consist of 6 to 8 students, and will be defined by EFREI inside “TP” groups,
- each team member must work on the business plan and on the prototype,
- each student’s “budget” is about one hundred hours **for technical aspects**,
- each student’s “budget” is about forty hours **for the business plan**,
- except for Internet server part where WEB technologies may be used, the programming language can be C++, Java or C#,
- the project should preferably be designed to run on Windows, but Unix-based solutions will also be accepted,
- for data management, a DBMS (data base management system) will be used; teams have to take care that their programming is DBMS independent, and as far as possible the project acceptance presentation should rely on Oracle,
- teams will use tools for managing code versions such as CVS or SVN.

5.3 Supplies delivery and mails

- each teacher will precise to its teams document delivery mode: electronic by mail or campus uploading, or printed copy in letter box,
- each team will have an official email address to receive instructions from their managers (if it is a group email such as googlegroups, take care that your managers can send you emails! Beware of spam-filters!)

5.4 Additional information

For any information, send an email to your technical or marketing manager, as soon as they are known.

You can also contact:

Object	Teacher	E-mail
Methods, project subject	Jean de Montcheuil	jean@montcheuil.com
Architecture, languages	Benoît Charroux	charroux@efrei.fr
Business plan	Andreas Topp	topp@efrei.fr

6 - Work to be done

6.1 Team organization and kick off

Teams will be defined by the teachers.

Before the kick off meeting each team has to:

- choose its project leader,
- choose a company name, a logo, a name for their software,
- define its organization for the first activities,
- discuss the main technical aspects that will be involved in the project's elaboration,
- set up its official email.

During the kick-off meeting with the technical and marketing managers:

- the team will present itself and its organization,
- there will be a common reading of project documents to clear up any misunderstanding,
- technical and marketing managers will give their instructions.

After this meeting, the team will write a report summarizing the team presentation and important points noted at the meeting. This report will be sent by email to the technical and marketing managers. This will allow validating email exchanges.

The team can then begin the product definition, which consists in elaboration of a first version of the Business Plan, and software preparation.

Note:

- all team members, including the project manager, have to work on the software development and on the business plan,
- for kick-off meetings to be productive, teams should have read project documents carefully, and have a vision of the project and their future software.

6.2 Business Plan redaction

At project start you have to give a first version of the Business Plan to your boss. After analyzing this document, your boss can decide to launch the project.

In this Business Plan, you have to:

- study the market and its environment,
- segment the market and define target customers
- define products and services to offer and the business model,
- define objectives,
- build a corporate strategy,
- define a schedule,
- examine juridical aspects,
- estimate financial needs and results.

The team can freely define the offer (products and services), objectives, corporate strategy and planning. The only constraints are that the initial offer is in domain of project subject, and that you shall start by the development of the prototype defined in project subject.

Notes on how to write a Business Plan will be given in management courses.

The first projects of version will be limited to key points of semester content; other aspects will be added in subsequent versions until the final version (V2) which will be complete.

6.3 Product definition

A preparatory study of the software will take place in parallel with the business plan's first version.

This study will allow to:

- precise the product content,
- establish a project outline,
- precise development method,
- examine possible technologies and needed resources (hardware and software tools),
- define the foreseen activities and foreseen content of each prototype,
- estimate development cost.

This step will be concluded by the delivery of the Product Definition Document (PDD).

The Product Definition Document (PDD) will follow the instructions of the PDD redaction guide and particularly contains:

- proposed content for the product,
- description of planned technical and architecture solution,
- development method,
- proposed technologies and needed means,
- general planning,
- realization foreseen activities and foreseen contents of successive prototypes,
- cost estimation

Product content

Generally statement of work does not define completely the product to realize. You have to give supplementary information to completely define the content of the product that you propose to realize.

Planned solution

Your planned technical and architectural solution cannot be a simple copy of the statement of work, but it must present the team's ideas on software content and how to organize the software and satisfy requirements. It must be sufficiently detailed to allow the definition of your needs and to estimate each task's workload.

Development method

You will use Scrum derived agile iterative development method. After definition steps, there will be one release realized in three sprints and the calendar (§ 5.1) is imperative.

Technologies and needed means

Possible technologies such as languages, workbenches, DBMS... will be examined and choices or possible choices will be defined, and means necessary to use these choices will be identified.

Planning

At this step, planning will be limited to general planning showing steps and milestones (see example appendix 1). Teams should use software tool such as MS Project to construct the planning.

More detailed diagram for steps will be done later at the beginning of the step.

Foreseen tasks and foreseen prototypes contents

Foreseen content of each sprint prototype will be defined, and a table will show the activities foreseen in each definition step and sprint in order to estimate workloads.

Cost estimation

Cost estimation will be based on workload estimation, and foreseen step activities.

6.4 Concept validation review

After product definition work, the team will present its proposals to technical manager and marketing manager: choice of application domain and market segment, services to offer, content of prototype to develop for demonstrations.

Proposals are validated by technical and marketing managers.

6.5 Software development

The software will be developed using Scrum derived agile method with a release definition step and the release realization in three sprints using layered Model View Controller (MVC) software architecture.

Imperative deliveries will be:

- Step Preparation Document (SPD) at the beginning of each step,
- General Requirements Specification (GRS) versions V1 and V2,
- Release Definition Document (RDD) versions V0, V1 and V2,
- Software Quality Plan (SQP) versions V1 and V2,
- User Manual (UM),
- Step Test List (STL) at the end of sprints 1 and 2,
- Acceptance Procedure (RAP) at the end of sprint 3,
- Code Integrated Documentation (CID) at end of project,
- Project Records (PR) at end of project.

More, some technology validation mocks up will be presented at end of release definition step and a prototype (P1, P2 and final) at the end of each sprint.

Mocks up content will be chosen by each team (for instance data base access, a view with refreshment mechanism...).

Deliveries are presented in appendix 4.

The imperative calendar and the deliveries for each iteration are indicated in §5.1.

6.6 End semester reviews

A review will take place at the end of first semester to validate work of semester, and an acceptance review take place at end of project.

For each of these reviews there will be a formal presentation by the team of its business plan and of content of its actual prototype.

For the first semester validation review, formal presentation must be in english.

Team will also do a demonstration of the functions of its prototype.

6.7 Debriefing

After acceptance, a debriefing with technical and marketing managers will take place. Thus you can share your experience of the project with other teams.

7 - Quality requirements

Team will present at project beginning dispositions adopted to ensure product quality

These dispositions will take into account all requirements of this document concerning quality:

- defining iterative development method,
- requirement analysis, GRS redaction,
- documents drafting (RDD, UM, RAP, PR...),
- using source management tools,
- project review at each step end.

Moreover, the following requirements will be taken into account:

- nomination of a quality manager for the project,
- installing a procedure for recording and solving problems,
- inspection of all delivered documents,
- realization of unit tests and integration tests.

Each project document or report will contain a first page or cover inspired from the first page of the present document bearing:

- company name,
- document title,
- identification number with form M1-type-nnn-company,
- edition number and diffusion date,
- name and signature of the author,
- name and signature of the project manager,
- name and signature of the quality manager.

8 - Project evaluation

There will be particular attention to:

- document readability (presentation and contents) and quality,
- quality of presentations,

- respect of deadlines,
- advancement between meetings
- team motivation and coherence,
- software quality,
- prototype presentation,
- quality plan,
- business plan content quality
- project management.

For each semester a grade will be given to team by technical manager for technical work, and a grade by marketing manager for Business Plan work.

Semester 1 evaluation criterions:

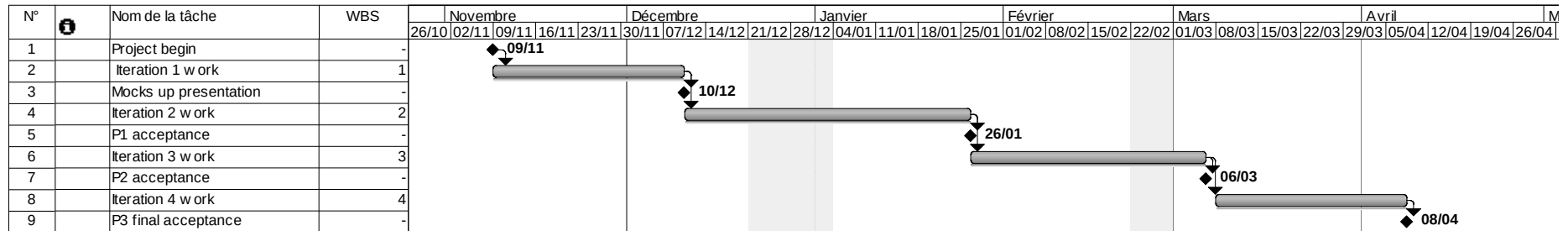
- BP:
 - o market study,
 - o market segmentation,
 - o target market analysis,
 - o definition of product and services,
 - o draft of semester 2 chapters,
- specifications:
 - o need coverage,
- architecture:
 - o domain analysis (Model layer),
 - o application analysis (Controller layer),
- quality plan application,
- implementation quality for the first prototype.

Semester 2 evaluation criterions:

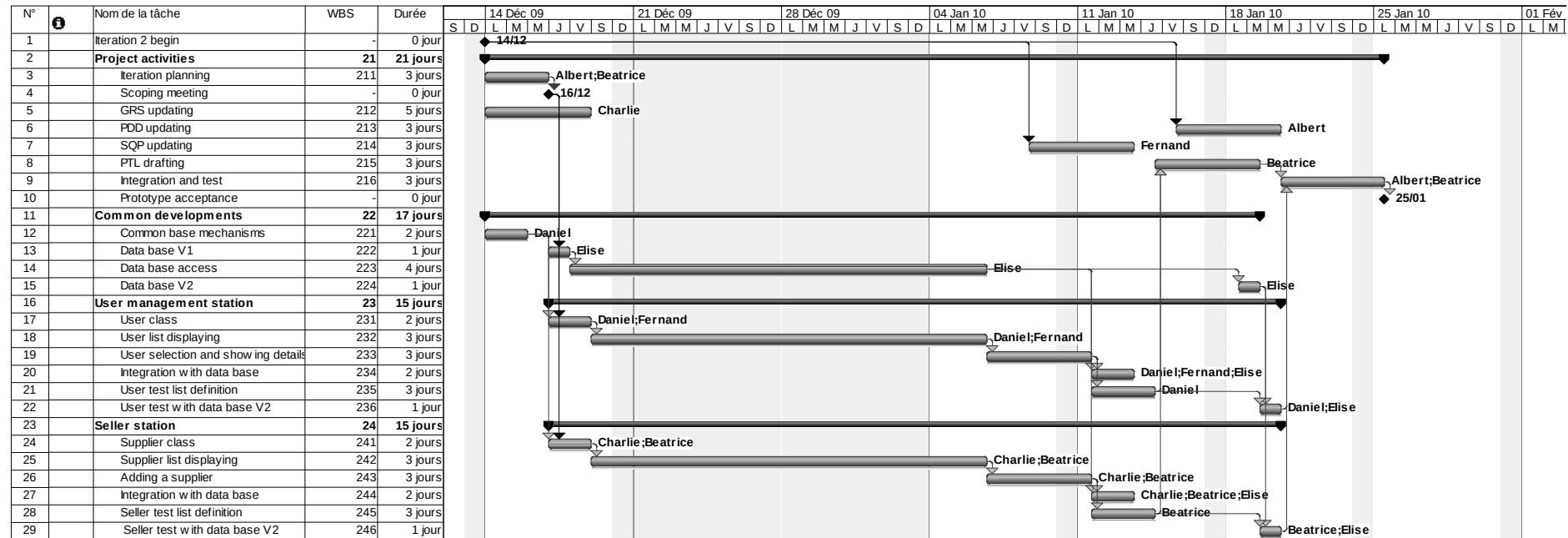
- BP:
 - o corporate strategy,
 - o marketing plan,
 - o finance,
- implementation quality:
 - o partition into layers,
 - o need coverage.
- quality plan application.

Each team member may have bonus or penalty taking into account his or her personal work and assiduity.

Appendix 1 General planning example



Appendix 2 Step planning example



Appendix 3 - Project leader

A2.1 Attributions

Project leader has to organize all project activities for product definition, Business Plan elaboration and software realization.

At project beginning and for definition steps and business plan works, the project leader is in charge of:

- defining project organization and planning,
- choosing managers,
- identifying tasks or activities to be done,
- identifying a single team member as responsible for each task or activity,
- identifying a single team member as responsible for each document.

In the other hand, for realization with scrum method, team is automanaged, task repartition is done collectively and reexamined at each "daily scrum". The project leader plays the roles of product owner and scrum master defined in scrum method.

During the project, the project leader is in charge of:

- verifying everyone executes their assigned tasks within scheduled delays,
- ensuring each document, each task, each software part will be finished within scheduled delays,
- organizing meetings to discuss technical aspects and difficulties or make decisions concerning project advancement
- ensure that everyone writes his/her individual activity report.

The project leader must also participate in software development and business plan elaboration.

A2.2 Advice

It is necessary that all team members participate in the choice of the project leader.

Choose a project leader who is very active and has good relational skills.

It is better if decisions are taken by consensus.

The planning will take account:

- holidays and planned absences (humanitarian trip...),
- extra-curricular activities,
- document rewriting,
- other school activities...

In case of problems occurring in team management, the project leader should first try to resolve them, and if he cannot, he must notify his technical or marketing manager immediately.

A2.3 Requirements

For each meeting with technical or marketing manager, the team must be complete.

If anyone has a major problem, they must notify before the meeting, and the team has to organize to compensate for the missing person.

Project leader must check that the official email address is valid, accessible by technical and marketing managers, and frequently read.

Appendix 4 Software development supplies description

A4.1 Step Preparation Document (SPD)

Each step preparation document will contain:

- an activity report of the precedent step (individual activity reports of each team member or synthetic table of their activities),
- a report of quality activities and quality problems during precedent step,
- a list of remaining problems at beginning of current step (problems noted during last prototype acceptance and other not solved problems) with decided actions,
- an advancement graph (burndown chart)
- for a sprint, the sprint backlog,
- a detailed planning of the step.

An example of detailed planning is in appendix 3.

An individual activity report model is in appendix 5.

A4.2 Project Records (PR)

Project Records will contain activity and quality reports for last iteration and a summary of useful data for future projects:

- project characteristics (area, complexity, number of source code lines...),
- work loads previewed and realized,
- used tools, appreciation of these tools,
- encountered problems and adopted solutions.

A4.3 General Requirements Specification (GRS)

GRS will contain a description of system mission and offered services or user scenarii.

More information is given in GRS drafting guide.

Version V1 will be complete, and V2 will be an update.

A4.4 Release Definition Document (RDD)

It contains the release backlog, presenting features (user stories coming from requirements and technical stories coming from architecture) and their tests (storytests), and diagrams showing interactions and dependences between stories or components, for instance:

- use case diagram,
- component diagrams,
- application classes diagrams,
- interaction diagrams,
- deployment diagram,
- database schemas.

More information is given in RDD drafting guide.

Version V0 will be preliminary, version V1 will be complete, and V2 will be an update.

A4.5 Software Quality Plan (SQP)

Its content will follow directives given in quality course, tailored for prototype development. It will contain at least:

- software project organization,
- organization to record and solve problems,
- development methods (limited to iterations definition and major activities),
- quality principles, and planned controls at each iteration,
- documentation management,
- configuration management.

More information is given in SQP drafting guide.

Version V1 will be complete, and V2 will be an update.

A4.6 Prototypes

Prototypes P1 and P2, and release prototype will be presented by the team in the computer labs.

The release prototype will be conforming to all requirements of the statement of work.

A4.7 User Manual (UM)

User manual will contain at least:

- software installation procedure,
- description of windows and menus,
- description of operator actions.

It may be a printed document or an on line help integrated to software.

More information on its content is UM drafting guide.

A4.8 Release Acceptance Procedure (RAP)

It contains:

- acceptance course,
- necessary means,
- release acceptance tests list (coming from release backlog storytests),
- release acceptance tests data.

More information on its content is in RAP drafting guide.

A4.9 Step Test List (STL)

It is a release acceptance procedure limited to the acceptance tests list.

A4.10 Code Integrated Documentation (CID)

Detailed design documentation will be done by comments integrated to source code.

These comments will follow all rules defined by automatic documentation tools for the used languages (following language Javadoc, Ndoc, Doxygen...).

There will be at least a comment cartridge for each class, comments cartridge for each function (function role, parameters and return), and a comment for each variable declaration.

Class cartridge will contain:

- role in software and main functions,
- version number and creation date,
- the author of the class,
- and for each update from the integration, origin, number and update description.

Appendix 5 - Individual activity report model

Individual activity report

Period	
Team	
Member	

Dates	Hours	Activities

Total	
-------	--

Remarks