Project Plan

PMoC
3/24/2003
Version 1.3.1

T-76.115 Project Plan
PMoC

Version	Date	Author	Description
1.3.1	24.3.2003	JF	Updated DE-phase plans
1.3	7.2.2003	JF	Documented plans for I3 phase and peer testing + some minor additions and modifications
1.2	2.12.2002	JF	Added time plan for I2 phase
1.0	28.10.2002	MJ,JF	Final changes
0.9	28.10.2002	MJ	Added future planning
0.7	28.10.2002	BF	Modified according to the customers comments.
0.6	27.10.2002	BF	Human resources + table.
0.5	25.10.2002	JF, AG, KH	Worked mainly on introduction, terminology and practices and tools.
0.1	17.10.2002	JF, BF, AG, KH	Draft

Table of Contents

1. Introduction

The introductory chapter describes the general environment the project will be implemented in, as well as the intended outcome of the project.
The most important stakeholders, and the value of the project to these stakeholders, should also be described briefly. It is also a good idea to mention what the involved parties have agreed concerning the rights to the project outcome.
The introduction also presents the structure of the project plan and the content of coming chapters. This is to make it easier to read the document and to grasp the concept of the plan in full.

1.1 Purpose and scope of project

This project is a computer science course at Helsinki University of Technology, which is carried out during fall 2002 and spring 2003. The purpose is twofold, partly for the students participating in the project to learn to use a real software process, and get experience from working in a somewhat bigger group, in this case seven persons, and interacting with a real customer. Partly for the customer to get a peace of software with as god as possible quality within the boundaries of the course, that supports the customers' business- and / or strategic goals.

1.2 The product and environment

1.2.1 The product

The aim is to create a prototype of a software product that can be used for creating process and automation flow diagrams that could be used with the simulation software that the customer uses. The new software is used to verify whether or not the GML-specification (Gallery Markup Language) together with graphics defined through SVG is a good solution for creating, storing and managing the configuration data related to simulation models. In addition, if time allows, to enhance the modeling capabilities towards a more intuitive use of the user interface.

1.2.2 The physical environment

The product will mainly be used by the researchers at VTT.

1.2.3 The hardware- and software environment

As in use of the customer, the target software environment will be Win32 platforms running on normal personal computers, with additional software as needed.

1.3 Rights to project outcome

As agreed upon in the contract that both parts will sign, the customer will get the owner rights of the project result, but the group will get complete rights to use, copy, reproduce, develop, distribute and sell the results of the project.
Reference: Contract document.

1.3 Document overview

First parts of the document mainly handles general issues concerning the project, as current solutions, feasibility, project organization, resources and it's goals, chapters two to six. In chapter eight project practices and tools are discussed, chapter nine gives a detailed picture of the different phases of the project, as far as planned. The rest of the document also handles more general issues like directives, education-, installation- and deployment plan.
This document is complemented by many other documents, naturally found in the references.

2. Terminology and definitions

Reference: Appendix A, Glossary.

3. Customer's current solution

VTT uses Apros[3] (Advanced Process Simulation Software) to model and simulate industrial processes. Apros supports real-time dynamic simulation of different industrial processes that can be designed and configured with Grades, a graphical user interface. Grades is used to design a simulation model by choosing process components (e.g. pipes, valves, pumps) from the model libraries, by using them to draw flow sheet diagrams and later by giving values for the properties of the components. It is also possible for the users to build their own component models using Grades. The simulation model can then be run in real-time (Apros as server) to simulate the behavior (e.g. pressure, fluid levels) of the modeled system (e.g. a part of a power plant).

4. Feasibility

4.1. Customer motivation

The prototype (the product of this project) is to be used to test the GML (Gallery Model Language) -specification in practice. The use of the GML (SVG, XML) could lead to a more device- and application-independent approach to configuration of process simulation models. The prototype could possibly be included in any future GML-based application. Secondly, if verification of the GML-SVG combination is successfully accomplished, development of an enhanced more intuitive user-interface will be part of the project.

4.2 Cost of the project

The project is to be implemented with 1400 man-hours from the group and about 60 hours from the customer. A theoretical hourly wage of 25€ is assumed (a software firm would probably charge more per hour but use less amount of hours instead). As it seems at an early stage of project (end of PP-phase, 25.10.2002) there will not necessarily be any need of acquiring any additional software.

Total cost: 1400h * 25€ + 60h * 25€ = 36500€

Since the project is a prototype and a part of a research project there isn't any negative effect or cost due to transition to a new system.

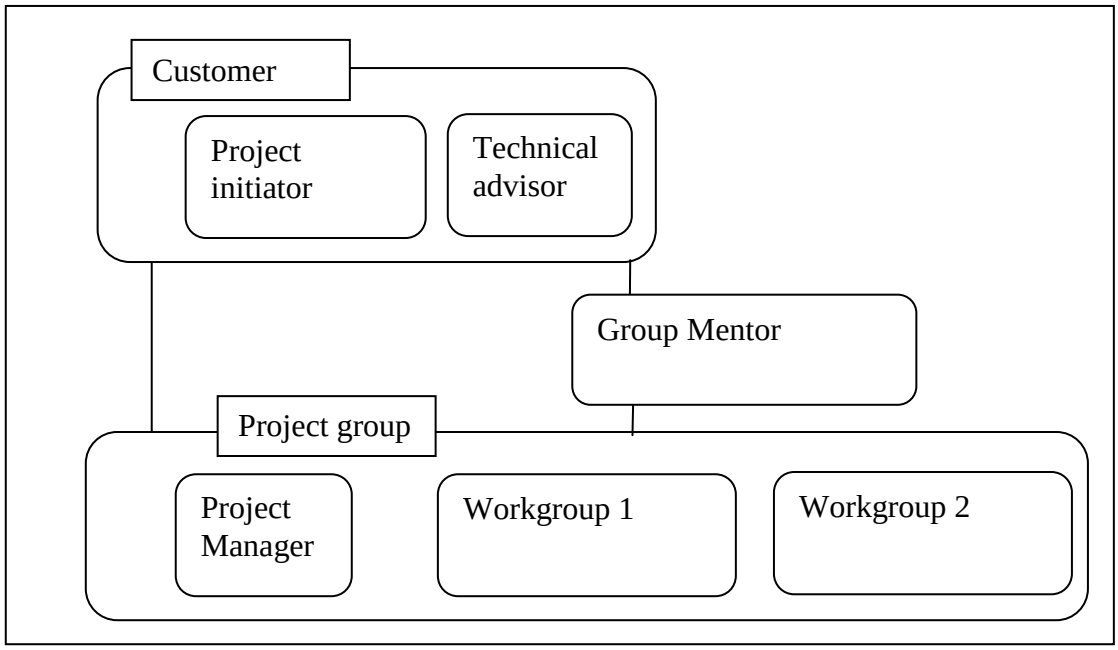
5. Project organization

5.1 Project group

5.1.1 General

The project group consists of seven people studying at Helsinki University of Technology, all at the department of Computer Science.
Group webpage is found atwww.yawc.net/T-76.115
Using email the group can be contacted at johan.forsback@hut.fi

5.1.2 Organization chart



5.1.3 Members presentation

Role	Developer
Name	Tomas Backas
Telephone	044-5558686
E-mail	tbackas@cc.hut.fi
Web-page	
Interests & skills	
Education & workexperience	

Role	System architect, tester
Name	Björn Forss
Telephone	050-5648320
E-mail	bforss@cc.hut.fi
Web-page	
Interests & skills	Graphics implementation, architecture design
Education & workexperience	5. year student at HUT. Most workexperience from Java-programming.

Role	Project manager
Name	Johan Forsbäck
Telephone	041-5272029
E-mail	johan.forsback@hut.fi
Web-page	
Interests & skills	Project management, architecture design, code development, Java with additional tools
Education & workexperience	4. year student at Helsinki University of Technology. Been involved in several commercial- and research- software projects during last three years.

Role	Developer, tester
Name	Anders Gebala
Telephone	050-5216555
E-mail	agebala@cc.hut.fi
Web-page	
Interests & skills	
Education & workexperience	

Role	Developer
Name	Kenneth Haglund
Telephone	050-5327028
E-mail	khaglund@cc.hut.fi
Web-page	
Interests & skills	
Education & workexperience	

Role	Assisting-, test-, and quality manager
Name	Markus Jakobsson
Telephone	050-3543188
E-mail	majakobs@cc.hut.fi
Web-page	
Interests & skills	
Education & work experience	

Role	System architect
Name	Jan Lönnberg
Telephone	
E-mail	jlonnber@niksula.hut.fi
Web-page	
Interests & skills	
Education & workexperience	

5.2 Stakeholders

Present the other stakeholders of the project (the customer, external consultants, etc) at a person level. Indicate at least the name, job description, and role in the project of people involved on the customer's side.

Role	Customer
Job description	Project manager for Gallery project, VTT
Name	Tommi Karhela
Telephone	050-5822984
E-mail	tommi.karhela@vtt.fi

Role	Technical advisor
Job description	Experienced user of similar programs as the one to produce
Name	Pasi Laakso
Telephone	
E-mail	pasi.laakso@vtt.fi

Role	Second technical advisor
Job description	
Name	Paavo Kotinurmi
Telephone	
E-mail	paavo.kotinurmi@hut.fi

Role	Third technical advisor
Job description	Works with net interface for connection to Apros database
Name	Jyrki Peltoniemi
Telephone	
E-mail	Paavo.kotinurmi@hut.fi

6. Project goals and end criteria

6.1 Goals of project group

The goals of the project group are to plan, create and document a system that fulfils the demands of the customer, thereby also fulfilling the groups own personal goals and at the same time the goals of the course attended. By doing this, the project group also especially expects and wishes to learn about the whole project process, and not only the possible extra lesson in programming practice, although the programming environment is new to the whole group. The project groups also wants to finish the project within the time set for the task, and therefore limit the project to a reasonable level of complexity.

6.2 Goals of customer

The goals of the customer are, in descending priorities among the two main priorities:

1. **Verification of GML**
   - 1.1. Verification of the graphical model on two hierarchical levels including references to the topological model
   - 1.2. Verification of the graphical and topological model on N hierarchy levels, with only static lifting of terminals.
   - 1.3. Verification of the graphical and topological model on N hierarchy levels, also with dynamic lifting of terminals.

2. **Post delivery development**
   - 2.1. Internal interfaces should be designed so that the query forms of the components can be developed later on.
   - 2.2. Internal interfaces should be designed to ease later adding of trend- and monitor data retrieving.
   - 2.3. Internal implementation logics and data structure should be designed to ease integration with the Galleria architecture later on.
   - 2.4. Rotation, flipping and snapping-to-grid of symbols should be easy to add later on.
   - 2.5. Animation properties should be possible to add later on.

6.3 Goals of project

The goals of the project are to create a system that follows as many of the customers wishes as possible, but concentrating on the most essential goals of the customer, being the verification of the GML -specification; either showing that it is a working design or pointing at the problems of the design - possibly also correcting possible such problems. The most central features of the system, being xml-handling and limited editing of both components and the actual connecting of components into a whole, are to be created. Any of the many possible additional features are a plus, but not necessary for the project outcome.

6.4 Project abort criteria

The project will be aborted if 3 or more members of the project group drops out. The project will also be aborted in the unlikely event that 90% of the available time has been used, and there is no end in sight of the project, and there is no way to cut down on demands and still get a working end product. The final decision to abort the project will be made by (what is left of) the project group alone. In the case of the premature termination the project group will transfer the knowledge, code and documentation already gained in the project to the customer’s organization.

6.5 Project end criteria

The project will end when both the project group and the customer is satisfied with the result. However this will be done in the limits of the course latest dead line of April 16th, 2003 or when the group has used up its resources of a total of 1400 man-hours for the project. A final meeting between the project group and its mentor and the customer will be held to establish the successful ending of the project. If, during the working phase, the project turns out to be hard or impossible to finish according to

the main and essential goals of the project, the project group and the customer will meet to discuss the possible cut-down of the requirements to allow the project to end instead of having to abort the project which should only be the last resort.

7. Project resources

7.1 Human resources

The human resources are divided equally for every phase of the project. Each group member should use about 200 hours for the whole project. All group members are fulltime students and have other courses that take time. Events like exams, tasks, and projects in other courses will affect the time when the group members can work for this project (see Appendix B).

Table 1: Planned effort - hours to be spent on project

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
PP	40	50	33	30	40	50	50	293
I1	40	45	47	40	46	46	44	308
I2	60	40	60	60	55	45	45	365
I3	40	35	40	50	40	40	40	285
DE	20	30	20	20	20	20	20	150
Total	199	200	200	200	201	201	199	1400

Table 2: Planned effort vs. realized as to 7.2.2003

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
PP-I2 plan	140	135	140	130	141	141	139	966
PP-I2 real	146	144	154	145	138	152	153	1032
Diff.	6	9	14	15	-3	11	14	66

7.2 Technical resources

The group can use the computer classroom A 218 in the computer science building for working on this project. The classroom is also used by other courses, so the availability is not known. The group may also have the possibility to use two computers at the customer's premises for this project

8. Project practices and tools

8.1 Requirements engineering

Reference: Requirements Documentation.

8.2 Project software tools

8.2.1 Java SDK: Java Software Development Kit by Sun

The Java SDK is the main and normal free Java compiler to be used in the project.

8.2.2 Emacs: Unix editor

Emacs is a powerful GNU Unix text editor which will be used for all text editing purposes.

8.2.3 JBuilder: Java Integrated Development Environment by Borland

JBuilder is Borland's IDE for developing Java applications, this will be used together with the Java SDK in particular for building the graphical user interface of the project.

8.2.4 Tirana: Web based time management tool by SoberIT

Tirana is a time managing system to be used for managing the time used by the group in the project.

8.2.5 Tidsmaskinen: Distributed calendar tool for project group

Tidsmaskinen (The time machine) is a web based tool that helps the group find the most suitable timeslots during the weeks, for the group meetings and other events that need to be coordinated.

8.2.6 Phorum: Web based communication forum for project group

The Phorum is a private discussion forum with possibility of document and file storage as well, easily accessed via the web.

8.2.7 MS Project: Project management tool

MS Project is software for planning resources and scheduling, also for exporting the work codes used in Tirana (mentioned above) to track working hours.

8.2.8 Rational Rose: UML design tool

UML design tool for creating necessary UML-diagrams needed for project documentation.

8.2.9 Ant: Java build tool

Ant will mainly be used for the regular builds described in document "Integration and regular builds".

8.2.10 CVS: Version control system

CVS is used as the repository for all code in the project. Reference: CVS Management document.

8.2.11 WinCVS: CVS tool for Windows

See CVS Management Document.

8.3 Design methods

8.3.1 Use cases

Reference: Use cases Document.

8.4 Programming

The project will be implemented in Java. Reference: Coding Convention Document.

The version management is done using CVS. Reference: Version Management Document.

Regular builds and integration will be used. Reference: Communication practices document

8.5 Testing practice

Reference: V-model testing document

8.6 General Work practices

8.6.1 Work environment

The group will use the computer classroom A 218 as major work environment. The technical resources are described in section 7.2. Due to the group-members very different weekly timetables, will a big part of the actual work be done individually or in smaller groups. This sets great demands on the communication inside the group.

8.6.2 Communication between the group members and meeting practices

Reference: Communication practices document

8.7 Documentation

All documentation is written in English. The documentation will be published in pdf-format. Pdf is good for making stylish documents. Pdf is also good for publishing documents on the web. The documents will be available on the project web-page.

9. Phasing

9.1 Project planning phase

24.9. - 31.10.2002

Investigation and planning of the whole project: meetings are held between the project group and the customer to create a clear picture of all requirements for the project. A Project plan and Requirement specification are written, which both stakeholders should acknowledge. Possible programming environment, as well as the external components or modules that are to be used in the project, are investigated thoroughly and chosen, in cooperation with the customer.

9.2 Implementation phase 1

1.11. - 5.12.2002

A main scaffold of the project should be made: a main system with certain set interfaces will be created, to allow for the three main different parts of the program to be distributed among the different subgroups within the project group. The themes for the subgroups are the XML-handling, the symbol editor and the model editor including a net tree view.

About 365h are allocated to this phase of the project. The resources are planned to be used as follows:

Table 4: Planned effort - hours to be spent on project, phase I1

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
Managing	0	10	0	0	2	3	0	15
Meetings	12	20	12	12	12	15	12	95
Req. & Spec.	3	5	5	3	2	5	5	28
Design	0	0	15	0	5	0	15	35
Implementation	15	0	5	15	10	5	5	55
Testing	4	0	5	2	10	8	2	31
Documentation	5	10	5	5	5	10	5	47
Total	39	45	47	37	46	46	44	304

Meetings

2-3 customer meetings	40h
3-4 group meetings (in smaller groups when possible)	45h
1 mentor meeting	10h

Requirements and specifications

The requirements are further refined with the customer.	13h (all)
System specifications are defined.	15h (all)

Design

Jan and Björn are the main system architects.	
The systems architecture will be designed	35h(JL,BF,AG)

Implementation

The core system functionality will be implemented.	
XML-handling	15h(BF,MJ)
Symbol editor	10h(JL,HK)
Model editor	10h(JL,TB)
Net tree view tool	10h(BF,AG)
Other modules	10h(all)

Testing

Markus is responsible for testing.	
Planning and managing testing	8h(MJ)
System testing planning and execution	5h (BF,MJ)
Integration testing planning and execution	8h (AG,MJ)
Unit testing	10h (all)

Documentation

The following documents will be written:	
Updated project plan	6 (all)
Updated requirements document	6 (all)
Technical specification	15 (all)
Test plan	10 (MJ,BF,AG)
Test report	5 (MJ,BF,AG)
Progress report	5 (JF)

9.3 Implementation phase 2

6.12.2002 - 13.2.2003

After implementation phase 2 the project should already be well off, with the main features on good way. The features with lower priority in each subgroup get added as the work up to that time get confirmed as correct by the customer.

Table 5: Up-to-date Effort and future plan

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
PP	37	52	34	29	39	54	55	300
I1	42	42	58	42	45	35	46	310
I2	50	50	50	60	50	47	43	350
I3	50	35	40	50	45	44	40	285
DE	20	30	20	20	20	20	20	150
Total	199	209	202	201	199	200	204	1414

About 350h are allocated to this phase of the project. The resources are planned to be used as follows:

Table 6: Planned effort - hours to be spent on project, phase I2

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
Managing	0	20	0	0	2	3	2	27
Meetings	10	20	10	10	10	10	10	80
Req. & Spec.	0	0	3	0	0	0	3	6
Design	0	0	8	0	5	0	8	21
Implementation	25	0	15	32	20	15	5	112
Testing	6	0	3	3	5	8	2	27
Debugging	4	0	3	10	3	1	3	24
Documentation	5	10	8	5	5	10	10	53
Total	50	50	50	60	50	47	43	350

Meetings

2-3 customer meetings	40h
3-4 group meetings (in smaller groups when possible)	30h
1 mentor meeting	10h

Requirements and specifications

The requirements are further refined with the customer.	3h
System specifications are refined.	3h

Design

Jan and Björn are the main system architects.

The systems architecture will be designed	21h(JL,BF, AG)

Implementation
System is implemented further, and new parts added

XML-handling	40h(TB,JL,AG)
Symbol editor	30h(KH,BG,MJ)
Model editor	32h(KH,BG,MJ)
Net tree view tool	10h(JL,AG)
Other modules	10h(all)

Testing
Markus is responsible for testing.

Planning and managing testing	5h(MJ)
System testing planning and execution	7h (Several)
Integration testing planning and execution	7h (Several)
Unit testing	10h (Several)

Documentation
The following documents will be written and updated:

Updated project plan and personal methods	6 (all)
Updated requirements document	4 (all)
Technical specification	12 (all)
Test plan	5 (MJ)
Test report	5 (MJ,)
Progress report	5 (JF)
Code documentation	17 (Several)

9.4 Implementation phase 3

14.2.- 27.3.2003

All features agreed on will be implemented, and possible half-done parts will be finalized. Peer testing as well as group internal testing will be accomplished. Documents are modified and added as required. Necessary preparations and planning of delivery-phase will also be done.

Table 7: Up-to-date Effort and future plan

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
Real PP	37	52	34	29	39	54	55	
Real I1	45	46	59	48	47	37	46	
Real I2	64	46	61	68	52	61	52	
Plan I3	39,5	40	29,5	34,5	39,5	38	33,5	
Plan DE	15	16,2	17	21	23	11	14	
Total	200,5	200,2	200,5	200,5	200,5	201	200,5	1403,7

The table above is up to date (7.2.2003) showing the realized hours for first three phases but the hours for phase I2 will grow a little, thus affecting DE phase hours (or final hours)

Table 8: Planned effort - hours to be spent on project, phase I3.

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
Studying	3	1	1	1	3	1	0	10
Meetings	5	10,5	9,5	5	7	9	8	54
Maintenance	1	0	0,5	1	1	0,5	0	4
Design	0	0	1	0	0	0	3	4
Implementation	21	0	12	18	22	11	3	87
Testing	6	0	3	4	3	12	7	35
Documentation	2,5	3,5	1,5	4,5	1,5	3,5	10,5	27,5
Management & organizing	1	25	1	1	2	1	2	33
TOTAL	**39,5**	**40**	**29,5**	**34,5**	**39,5**	**38**	**33,5**	**254,5**

All estimates, both for table 8, and the list below, are based on the hours reported during phase I3, then somewhat adjusted, taking into account some of the differences between phases I2 and I3. E.g. peer testing, more focus on implementation and testing, and less on meetings and design, to mention some.

9.4.1 Detailed plan for I3 phase

Explanations:

- GG = graphics group, consisting of BF, MJ and KH.
- / means 'except for …'

Studying		
Java & Swing	3	TB,KH,AG,MJ
Batik / SVG	2	BF,MJ
GML / XML	4	TB,AG
Project Management etc.	1	JF
Project related	0	
Total	10	
Meetings		
Group meetings	26	all
Customer meetings	20	all
Mentor meeting 5	8	all
Total	54	
Maintenance		
Installations	4	TB,BF,KH,AG,MJ
Other comp. maintenance	0	
Total	4	
Design		
General system design	3	BG,JL
Subsystem division and their relationships	0	
System architecture design	1	JL
Total	4	

Implementation		
XML-handling (GML)	15	TB,AG
IO-handling	10	TB,AG
Browser	10	AG
Symbol Editor	8	GG
Net Editor	15	GG
Coordinator	5	BF,KH,AG
GUI	10	GG,AG
Other modules (Misc)	11	all /JF
Rework to other features	3	KH,MJ,JL
Total	87	
Testing		
Planning the testing	5	MJ
Running the tests	15	all /JF
Test reporting	7	all /JF
Peer testing	8	TB,MJ,JL
Total	35	
Documentation		
Writing/Updating the technical specification	6	JL
Writing/Updating requirements document	2	JL
Documenting use of personal methods	3,5	all
Misc Documentation	7	all
User Guide	3	KH
Writing the progress report	2	JF
Peer testing documentation	4	TB,MJ,JL
Total	27,5	
Management & organizing		
Organizing the project	16	JF,AG,JL
Communicating schedule & plans	4	JF
Detailed planning of the next phase	3	JF
Preparing for the project review	3	JF
Project review	7	All
Total	33	
TOTAL	254,5	

9.4.2 Peer testing

The PMoC group will accomplish the peer testing for the OpenLogbook software project 24.2 – 3.3. returning the peer test documents at the latest 3.3. The early dates at request of the OpenLogbook team.
The OpenLogbook will test the proconf software 3.3-10.3 and return their documentation of the peer tests at the latest 10.3.
Necessary documentation and the software of course, will be exchanged no later than the Friday before the respective testing weeks. If needed a meeting to debrief the peer testing is planned for week 11, where the testers and possibly a some other people from the groups will be present, anyway not the whole groups.
More information about the peer testing is found in the Test Plan chapter 3.3.1.

9.5 Delivery phase,

28.3.2003 - 16.4.2003

During this phase the code is finished, all bugs possible fixed and a working installation package should be made. The documents will be finished and final delivery to the customer made.

	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	Total
Real PP	37	52	34	29	39	54	55	
Real I1	45	46	59	48	47	37	46	
Real I2	64	70	61	68	52	61	52	
Real I3	39	16	41	50	45	35	30	
Plan DE	20	20	10	10	23	18	20	
Total	205	204	205	205	206	205	203	1433

Table 9: Up-to-date Effort and future plan (24.3.03)

Exact time estimations for individuals and different tasks is not possible to make as accurate as the I3 phase, which was based on the earlier implementation phases.

10. Standards, directives, and orders

The project members are to remember that everything seen, heard or tasted or even touched at the customer's premises is kept confidential.

11. Risk management plan

See different Risk management document.

12. Education plan

12.1 Group internal education plan

The group is not familiar with all techniques and concepts needed for the project. The group will need education in using the tools used in the project and in the techniques used. This education will mostly be self-study. At the beginning of the project a training day when the group was familiarized with GML was held at the customers premises.

12.2 Customer education plan

A demonstration in how the system works will be held at the end of the project for the customer. Customer may want to see draft versions of the system already earlier. In this case demonstrations and code inspections can be arranged also between project phases.

13 Installation plan

At the end of the project the system, all documentation and source codes will be delivered to the customer on CD-ROM. A demonstration of the system will be held as described in 12.2. The project group has no responsibility for the outcome of the project after the project is finished. For more support, an agreement between the group and the customer has to be made.

14 Deployment plan

The system will not replace the old system. The system will be taken into use beside the old system. The customer has to further develop the system before it is ready to replace the old system. The system developed in this project is a prototype for proving a concept that the customer has. Installation and delivery will be done as described in section 13.

References

1. Legal contract, VTT lawyer © 2002
2. Version Management Document, Jan Lönnberg © 2003
3. Risk management plan, Johan Forsbäck © 2003
4. Apros web page http://www.vtt.fi/aut/tau/ala/apros.htm
5. Use cases, Björn Forss © 2003
6. Requirements Document, Jan Lönnberg © 2003
7. Technical Specification, Jan Lönnberg © 2003
8. Test Plan, Markus Jakobsson © 2003
9. V-model testing, Markus Jakobsson © 2003
10. Communication and meeting practices, Kenneth Haglund © 2003
11. IO Specification, Jyrki Peltoniemi © 2003
12. Integration and regular builds, Anders Gebala © 2003
13. Coding conventions, Tomas Backas © 2003

Appendices – Table of Contents

Appendix A - Glossary

SVG	Scalable Vector Graphics 1.0 is a language for describing two-dimensional vector and mixed vector/raster graphics in XML
XML	Extensible Markup Language is a simple, very flexible text format derived from SGML (ISO 8879). Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere.
GML	Gallery Markup Language, an XML based model configuration specification developed at VTT.
HUT	Helsinki University of Technology.
SoberIT	Software Business and Engineering institute at HUT.
UML	The Unified Modeling Language (UML) is a language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.
CVS	CVS is the Concurrent Versions System, the dominant open-source network-transparent version control system.
VTT	Technical Research Centre of Finland.
PDF	Portable Document Format. Open de facto standard for document distribution. Reference: http://www.adobe.com/products/acrobat/adobepdf.html

Appendix B – Detailed view of the efforts for phase I3.

MS-P	Topics	Compl. real	plan I2	real I2	diff I2	new plan I3	Tomas Backas	Johan Forsbäck	Björn Forss	Kenneth Haglund	Anders Gebala	Markus Jakobsson	Jan Lönnberg	
I3-ST	**Studying**													
	Java & Swing		0	5	5	3,0	1,0			1,0	1,0			3,0
	Batik / SVG			0		2,0			1,0			1,0		2,0
	GML / XML		0	5	5	4,0	2,0				2,0			4,0
	Project Management etc.			0		1,0		1,0						1,0
	Project related		0	0,5	0,5									0,0
	Total	**9**		10,5		10,0	3,0	1,0	1,0	1,0	3,0	1,0	0,0	10,0
I3	**Meetings**													
	Group meetings		30	61,55	31,55	26,0	3,0	5,0	4,0	3,0	4,0	4,0	3,0	26,0
	Customer meetings		40	36,5	-3,5	20,0	1,0	4,0	4,0	1,0	2,0	4,0	4,0	20,0
	Mentor meeting 5		10	10,75	0,75	8,0	1,0	1,5	1,5	1,0	1,0	1,0	1,0	8,0
	Total	**100,3**		108,8		54,0	5,0	10,5	9,5	5,0	7,0	9,0	8,0	54,0
I3-SU	**Maintenance**													
	Installations			4,75		4,0	1,0		0,5	1,0	1,0	0,5		4,0
	Other comp. maintenance			0										0,0
	Total	**4,75**		4,75		4,0	1,0	0,0	0,5	1,0	1,0	0,5	0,0	4,0
I3-TD	**Design**													
	General system design			0		3,0			1,0				2,0	3,0
	Subsystem division and their relationships			0		0,0								0,0
	System architecture design		0	1	1	1,0							1,0	1,0
	Total	**12,5**		1		4,0	0,0	0,0	1,0	0,0	0,0	0,0	3,0	4,0
I3-IM	**Implementation**													
	XML-handling (GML)		40	39,5	-0,5	15,0	7,0				8,0			15,0
	IO-handling			0		10,0	10,0							10,0
	Browser		10	28,5	18,5	10,0					10,0			10,0
	Symbol Editor		25,6	112,1	86,5	8,0			2,0	4,0		2,0		8,0
	Net Editor		25,7	7,5	-18,2	15,0			5,0	5,0		5,0		15,0
	Coordinator			0		5,0			1,0	2,0	2,0			5,0
	GUI					10,0			3,0	3,0	1,0	3,0		10,0
	Other modules (Misc)		26,8	9,5	-17,3	11,0	4,0		1,0	3,0	1,0		2,0	11,0
	Rework to other features		0	4,5	4,5	3,0				1,0		1,0	1,0	3,0
	Total	**189,6**		201,6		87,0	21,0	0,0	12,0	18,0	22,0	11,0	3,0	87,0
I3-TE	**Testing**													
	Planning the testing		4	9	5	5,0						5,0		5,0
	Running the tests		20	8,45	-11,6	15,0	2,0		2,0	3,0	2,0	3,0	3,0	15,0
	Test reporting		5	7,2	2,2	7,0	1,0		1,0	1,0	1,0	2,0	1,0	7,0
	Peer testing			0		8,0	3,0					2,0	3,0	8,0
	Total	**34,45**		24,65		35,0	6,0	0,0	3,0	4,0	3,0	12,0	7,0	35,0
I3-TD	**Documentation**													
	Writing/Updating the technical specification		5	18,65	13,65	6,0							6,0	6,0
	Writing/Updating requirements document					2,0							2,0	2,0
	Documenting use of personal methods					3,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	3,5
	Misc Documentation		30	11,1	-18,9	7,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	7,0
	User Guide					3,0				3,0				3,0
	Writing the progress report		10	0,5	-9,5	2,0		2,0						2,0
	Peer testing documentation			0		4,0	1,0					2,0	1,0	4,0
	Total	**30,45**		30,25		27,5	2,5	3,5	1,5	4,5	1,5	3,5	10,5	27,5
I3-MA	**Management & organizing**													
	Organizing the project		0	23,05	23,05	16,0		14,0			1,0		1,0	16,0
	Communicating shedule & plans					4,0		4,0						4,0
	Detailed planning of the next phase		10	2,2	-7,8	3,0		3,0						3,0
	Preparing for the project review		2	6,75	4,75	3,0		3,0						3,0
	Project review		7	5,25	-1,75	7,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	7,0
	Total	**27,25**		37,25		33,0	1,0	25,0	1,0	1,0	2,0	1,0	2,0	33,0
TOTAL		**408,3**		**418,8**		**254,5**	**39,5**	**40,0**	**29,5**	**34,5**	**39,5**	**38,0**	**33,5**	**254,5**
							148,8	143,8	155,7	144,8	137,8	151,3	153,8	
Tot after I3							188,3	183,8	185,2	179,3	177,3	189,3	187,3	
Left							11,8	16,2	14,8	20,8	22,8	10,8	12,8	