

Timo Poranen (ed.)

Software Projects 2006-2007



DEPARTMENT OF COMPUTER SCIENCES
UNIVERSITY OF TAMPERE

D-2007-7

TAMPERE 2007

UNIVERSITY OF TAMPERE
DEPARTMENT OF COMPUTER SCIENCES
SERIES OF PUBLICATIONS D – NET PUBLICATIONS
D-2007-7, JUNE 2007

Timo Poranen (ed.)

Software Projects 2006-2007

DEPARTMENT OF COMPUTER SCIENCES
FIN-33014 UNIVERSITY OF TAMPERE

ISBN 978-951-44-6993-0
ISSN 1795-4274

Preface

All projects of the academic year 2006-2007 are finished now. This report contains project stories of all 18 software projects and two usability teams. The stories give an overview of the project, with positive and negative experiences, and statistics.

Table 1 contains course statistics starting from academic year 2002. Some numbers from earlier courses are unknown; these are marked with a question mark (?).

Table 1: Course statistics 2002-2007.

Academic year	Projects	Usability teams	PW students	SPM students	Average project size
2002-3	15	0	?	?	?
2003-4	15	0	?	?	?
2004-5	13	0	60	7	?
2005-6	19	1	98	8	1008 hours
2006-7	18	2	87	34	1089 hours

Table 2 gives an overview of the projects of this year. The table lists project's name, project's type (WWW = a WWW application often containing a database, Mobile = a Mobile phone application or Appl. = a stand-alone application), client (Dept. = Department of Computer Sciences, Univ. = other university units, Gov. = non-profit governmental organisations, or Company), used software development model (WF = Waterfall, INC = Incremental, GUIDe = Graphical User Interface driven Development, EVO = Evolutionary, ITE = Iterative), group size (Number of managers + size of the project team + usability experts), and total working hours of the project.

During the course project managers kept up the working hours of the project team. The working hours are divided into nine activities: project planning, project management, requirements specification, software design, code (programming), integration and testing, reviews, repair (bug fixing) and other. All projects, except usability teams, give their working hours divided into these categories.

The statistic sections of almost all projects contain tables for general project information, requirements and high-level design outcome, design outcome, projects documents, codelines, and productivity metrics.

I thank our clients for their interesting project topic suggestions. And

Table 2: General project statistics.

Project	Type	Client	Dev. Model	Group	Hours
HAT	WWW	Dept.	INC	1+3+1	878
Mixel	Appl.	Univ.	INC	2+5+1	1503
SPOT	WWW	Dept.	ITE	2+5+1	1138
Rogue	Mobile	Dept.	INC	2+4+1	1566
Moreeni	WWW	Univ.	EVO	2+5+1	733
Tikir	WWW	Gov.	WF	2+5+1	1196
Sitemaps	WWW	Company	WF	2+5+0	626
Modulo	WWW	Univ.	INC/WF	2+4+1	1392
ToVe	WWW	Company	WF/INC	2+4+1	1088
Koodisto	WWW	Gov.	WF/GUIDe	2+3+1	1046
EDict	WWW	Univ.	WF	2+4+1	1038
OurTown	Mobile	Company	WF/INC	2+3+1	846
FotTheWok	WWW	Company	EVO	2+5+1	759
SysMLL	WWW	Univ.	INC	2+2+1	1080
USVA	Appl.	Dept.	INC	2+5+1	1719
IOI2006	WWW	Dept.	INC/WF	0+4+0	765
CaleSync07	Mobile	Company	INC	1+5+1	922
Dice Corpus	WWW	Univ.	ITE	2+4+1	1421
Uteam-FI	All	Dept.	All	1+7	1373
Uteam-EN	All	Dept.	All	1+5	1137

finally, special thanks goes to all project work and software project management course students, you did it!

Tampere, June 2007
Timo Poranen

Contents

1	Usability Team - FI	1
1.1	Overview	1
1.2	Organisation and management	1
1.3	Methods and tools	2
1.4	Conclusions	3
1.5	Statistics	4
2	HAT 2006	6
2.1	Overview	6
2.2	Organisation and management	6
2.3	Methods and tools	7
2.4	Project phases	7
2.5	Conclusions	7
2.6	Statistics	7
3	Mixel	11
3.1	Overview	11
3.2	Project organisation	11
3.3	Project management	12
3.4	Methods and tools	12
3.5	Project phases	13
3.6	Conclusions	14
3.7	Statistics	14
4	SPOT	18
4.1	Yleiskuvaus projektista	18
4.2	Projektiorganisaatio	19
4.3	Työskentelymetodit ja työkalut	20
4.4	Projektin vaiheet	20
4.5	Yhteenvedo	21
4.6	Tilastoja	21
5	Rogue	25
5.1	Overview	25
5.2	Organisation and management	26
5.3	Methods and tools	26
5.4	Project phases	28
5.5	Conclusions	28
5.6	Statistics	29

6	Moreeni	32
6.1	Yleiskuvaus ohjelmasta	32
6.2	Organisation and management	32
6.3	Methods and tools	32
6.4	Project phases	33
6.5	Conclusions	33
6.6	Statistics	33
7	TIKIR	38
7.1	Yleiskuvaus	38
7.2	Projektiorganisaatio	38
7.3	Projektin eteneminen	38
7.4	Projektin hallinta	39
7.5	Koodin määrä	39
7.6	Ruutukaappauksia järjestelmästä	39
7.7	Tilastot	41
8	Sitemaps	44
8.1	Yleistä	44
8.2	Projektiryhmän jäsenet	44
8.3	Käytetyt välineet	45
8.4	Projektin vaiheet	46
8.5	Johtopäätökset	47
8.6	Tilastoja	47
9	Modulo	50
9.1	Yleistä	50
9.2	Projektiorganisaatio	50
9.3	Käytetyt välineet	52
9.4	Projektin vaiheet	52
9.5	Johtopäätökset	53
9.6	Tilastoja	54
10	ToVe	57
10.1	Yleistä	57
10.2	Projektiorganisaatio	57
10.3	Metodit ja työkalut	58
10.4	Projektin vaiheet	58
10.5	Johtopäätökset	59
10.6	Tilastoja	60

11 Koodisto	63
11.1 Yleiskuvaus ohjelmasta	63
11.2 Projektioorganisaatio	63
11.3 Projektin eteneminen	63
11.4 Johtopäätökset projektista	64
11.5 Projektin hallinta	65
11.6 Käytetyt välineet ja menetelmät	65
11.7 Tilastot	66
12 English Usability Team	68
12.1 Overview	68
12.2 Organisation and management	68
12.2.1 English usability team members	69
12.2.2 Group meetings	69
12.2.3 Weekly reports	70
12.3 Methods and tools	71
12.3.1 Methods	71
12.3.2 Tools	72
12.4 Conclusions	73
12.4.1 Experiences from the project	73
12.4.2 What to do better next time	73
12.5 Statistics	73
13 Electronic Dictionary Project	75
13.1 Overview	75
13.2 Organisation and management	75
13.3 Methods and tools	75
13.4 Project phases	77
13.5 Conclusions	78
13.6 Statistics	78
14 OurTown	82
14.1 Overview	82
14.2 Organisation and management	82
14.3 Methods and tools	83
14.4 Project phases	85
14.5 Conclusions	85
14.6 Statistics	86

15	Fot the Wok	89
15.1	Overview	89
15.2	Organisation and management	90
15.3	Methods and tools	90
15.4	Project phases	91
15.5	Conclusions	92
15.6	What to do better next time	92
15.7	Statistics	93
16	Extension Of SysMLL	96
16.1	Overview	96
16.2	Organisation and management	96
16.3	Methods and tools	96
16.4	Project phases	97
16.5	Conclusions	97
16.6	Statistics	98
17	USVA	103
17.1	Overview	103
17.2	Organisation and management	104
17.3	Methods and tools	104
17.4	Project phases	105
17.5	Conclusions	107
17.6	Statistics	107
17.7	Acknowledgement	107
18	IOI2006	111
18.1	Overview	111
18.2	Organisation and management	111
18.3	Methods and tools	112
18.4	Project phases	112
18.5	Conclusions	113
18.6	Statistics	114
19	CALESYNC07	116
19.1	Overview	116
19.2	Organisation and management	116
19.3	Methods and tools	116
19.4	Project phases	117
19.5	Conclusions	117
19.6	Statistics	118

20 DICE Corpus	122
20.1 Overview	122
20.2 Organisation and management	122
20.3 Methods and tools	123
20.4 Project phases	124
20.5 Conclusions	125
20.6 Statistics	125

1 Usability Team - FI

1.1 Overview

The Usability Team 2006 - 2007 was split up into two groups according to language. This report concerns the Finnish language group, abbreviated to UTeam-FI.

The Usability Team 2006 - 2007 was developed based on previous academic year's Usability Team (2005 - 2006). The Finnish group finally consisted of 7 members (one changed during the course due to workload distribution), while the English group had 5 members. In addition to these, both groups had their own project manager.

The idea of the Usability Team is that the team members participate into other projects to help as usability experts. In addition, the Usability Team takes care that each project's usability needs are met. The Usability Team chooses how the members are distributed into projects mostly based on the workload estimation of the projects. The Finnish group had 9 projects that were assigned a member, so some members worked on multiple projects.

1.2 Organisation and management



Figure 1: Project UTeam-FI.

The projects members of UTeam-FI were: Eveliina Houni, Petri Laitinen, Kalle Laukkonen, Virpi Pasto, Olli Savolainen, Jenny Sihvo and Taika

Tulonen. The project manager was Ilari Kajaste. (Missing from the picture are Eveliina, Virpi and Jenny).

UTeam prioritizes working in projects, so we held UTeam-FI meetings in irregular phases. We decided the aim should be to hold them about biweekly. Total of 16 meetings was held in the 32 weeks of the projectwork course, so the aim was met - although it should be mentioned there was a somewhat undesired low period during the spring. For Usability Team, there wasn't that much internal organizing required as the priority work was done withing other projects, but the coordination was of course left to the project manager.

The project was managed based on common group consensus rather than direct dictation of rules. In practice, the project manager could decide how to handle most issues, but the consensus approach provided the support of the whole team was behind the project manager.

In addition to working in their own projects, the Usability Team members were also recruited for testing and analyzing of usability issues for each other's projects. One of the key features of Usability Team was the peer support of other usability oriented people provided by the Usability Team concept.

In addition to team coordination, the project manager also developed the Usability Team concept further by writing templates and other documents. This should reduce the workload of future Usability Team managers. The project manager also kept the Usability Team website updated, and developed an automated hour reporting system suitable for the Usability Team. The Usability Team website is located at: <http://www.cs.uta.fi/~uteam/>.

1.3 Methods and tools

Usability Team Finnish group used the following tools during the project:

- Heuristic Evaluation (2)
- Peer Review (5)
- Usability Testing (2)
- Paper Prototype Testing (3)
- User Interface Interviews (1)

Of these, the Peer Evaluations were found to be very useful. The idea in a Peer Evaluation is that someone to presents an user interface, and then the group together informally analyzes the interface. This helps the presenter to find out what's working well and what's not, and get more ideas because

of the boarder perspective of the group. The focus should be on analyzing the interface, not just critiquing the design decisions made. To keep focus, heuristic lists or use cases can be used within the evaluation. A report of the findings should be written by the presenter.

Other useful method was naturally the Usability Testing, whether with paper prototypes or not. Usability Testing provides an unique view into the minds of the users and how they will interact with the interface.

The following types of usability related documents were produced:

- Usability Analysis (9, 108 pages)
- User Interface Plan (7, 100 pages)
- User Interface Evaluation (1, 7 pages)
- User Interface Screen Map (3, 8 pages)
- Usability Testing Plan (4, 29 pages)
- Usability Testing Report (4, 55 pages)
- Contextual Analysis (1, 25 pages)
- Paper Prototype (3)

Of these the Usability Analysis and User Interface Plan were required for most of the projects, and they also proved to be most important.

1.4 Conclusions

From the perspective of the Finnish group the problems the Usability Team faced this year were related to coordination with other projects. The biggest problem was with the complex nature of the interaction between UTeam and projects. As further development, the roles which Usability Team members can take should be more clearly defined, and the Usability Team members should be aware of these roles and general project work structure before being introduced to the project teams.

Despite the problems, UTeam-FI was thought of as an useful resource by the projects, and the members did much work for the projects with good results. The members of UTeam-FI also thought the Project Work course in Usability Team was educational experience, and especially a very good way for students of Interactive Technology to work in the project work course. The project UTeam-FI is then considered a success.

1.5 Statistics

UTeam-FI did 1372,5 working hours in total, and where 1132,5 was done by the members. Usability team used it's own working hour classification system, where hours were divided into four categories: "Meeting" for meetings about communicating, "testing" for performing usability tests, "course" for course, not project matter (lectures and personal reports) and "other" for any other work on projects. Figure 2 shows the distribution of these hours and a total.

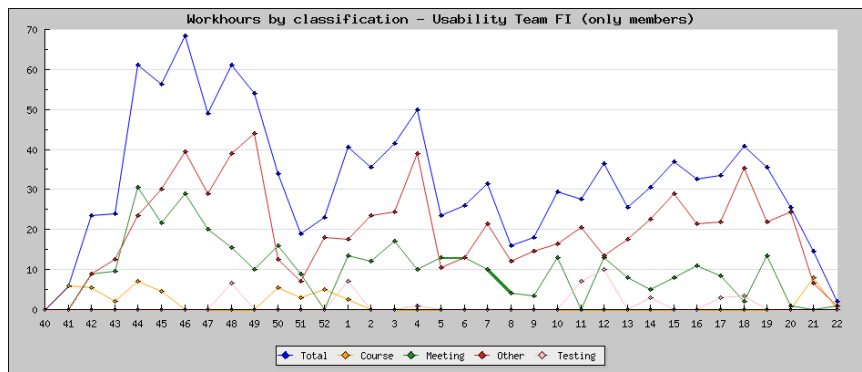


Figure 2: Weekly workload of the project members.

The work of Usability Team was to be concentrated on work for projects, not Usability Team itself. This also happened in practice, as can be shown in Figure 3]:

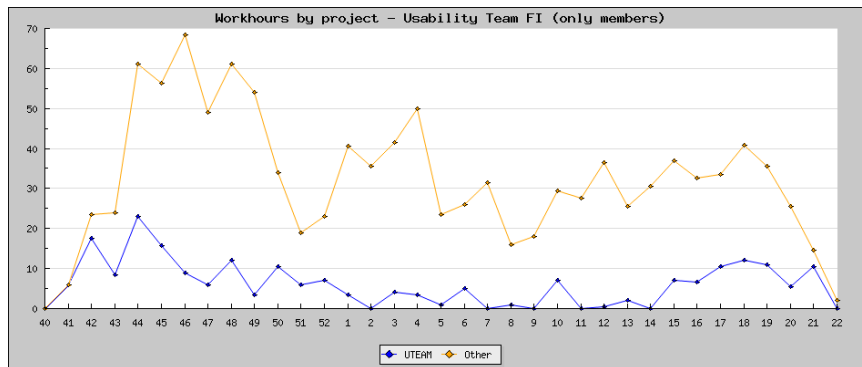


Figure 3: Workload of project members, distributed to projects.

The project manager is excluded from the graphs. For the project manager only a small amount of the work was done for other projects than Usability Team - to be exact 34 hours of the total 240 hours was for other than

Table 3: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team size)
1+7	N/A	19.10.2006	21.5.2007	215	1372,5	0,91

Table 4: Project's documents.

Document	Pages	versions
Preliminary analysis	5	1
Final report	26	4
Final story	4	1
Weekly reports	28	
Monthly reports	1	
Total	64	7

Usability Team, and most of these, 16,5 were done for Rogue project as a subproject where UTeam-FI gathered voice samples for the project.

2 HAT 2006

2.1 Overview

HAT 2006 project continued the development of the software done by the project HAT 2005. The HAT software is a support system for heuristic evaluation. Project's two primary goals were to implement missing features and test and improve the usability of the system. HAT is a WWW-based application written in Java.

2.2 Organisation and management

The client of the project was the Usability Laboratory of the University of Tampere.

The client's representatives were

- Saila Ovaska
- Harri Siirtola

The manager of the HAT 2006 project was

- Laura Vuorenoja

The members of the project group were

- Ilari Laitinen
- Juha Ollikainen
- Jussi-Heikki Uusitalo
- Virpi Pasto (from the usability group)

Ilari Laitinen and Juha Ollikainen held the primary responsibility for software designing and implementing. Virpi Pasto along with Jussi-Heikki Uusitalo tested the system and designed some user interface changes. Laura Vuorenoja dealt mainly with project management issues. Two of the project's members quit the project during the course.

The group had weekly meetings and the decisions made in them were afterwards emailed to group's members. In addition, the meeting minutes and all the other project documentation was available in the group's internet forum. The group communicated also by email and had its own IRC channel.

Reviews were held almost after each increment. The reviews were useful, the group got important feedback from the lecturer and the client. Both the documents and the system itself were reviewed in these meetings.

2.3 Methods and tools

The project group had to use the implementation techniques the previous group (HAT 2005) had chosen. PostgreSQL is the system's database and the data is manipulated with the help of Hibernate objects. In addition, Spring framework is used in building the system's architecture. The group thought that the implementation techniques were complicated and it took a lot of time to study their use.

The group used Eclipse and Subversion (Subclipse) to help the coding and version management. The Subversion repository was held in a virtual server that was provided by the CS Department. The version management worked well, and it was helpful that all the project stakeholders could use the system in the internet.

2.4 Project phases

In the beginning of the project the group spent a lot of time to getting to know the system and trying to figure out the actual requirements. However, the requirements changed during the whole project because new bugs were found all the time. Because of the bugs, the implementation of the new features and the usability testing delayed.

The implementation phase followed basically an incremental development model. Because of the delays the contents of the increments changed a lot from the initial plans. In addition, the quitting of one software developer had an effect on the schedule and increased other members' workload. The incremental model suited nevertheless well to the project, at least better than the waterfall model would have suited.

The main implemented functionalities (in addition to bug fixing) were the worklist that helps the user to follow a specific comment (Figure 4), the states of the comments (Figure 5) and the possibility to merge a comment to another comment (Figure 6).

2.5 Conclusions

The project was challenging, but despite all the difficulties the group was satisfied with the outcome. The advise for the other groups that continue previous year's project group's work is not to trust at all old documentation and test results :)

2.6 Statistics



Figure 4: Worklist in the case view.



Figure 5: Information about the comments in the component view.



Figure 6: A merged comment.

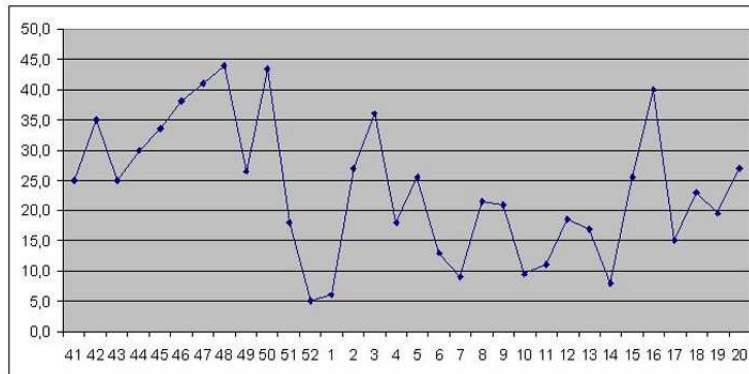


Figure 7: Weekly working hours.

Table 5: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
1+3+1	INC	9.10.2006	22.5.2007	226	878,25	0,8

Table 6: Group effort by activity.

Activity	Plan-ning	Man-nage-ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	39,5	52,5	54	48	187,5	26	23	2	323	755,5
%	5,2	6,9	7,1	6,4	24,8	3,4	3,0	0,3	42,8	
Usab.										122,75
Total										878,25

Table 7: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
49	7	4	3	0	0

Table 8: Project's documents.

Document	Pages	versions
Preliminary analysis	10	2
Project plan	30	3
Requirements specification	36	2
Architecture plan	10	3
Design plan	9	4
Project's usability test plan	8	1
User usability test report	8	1
Test plan	44	2
Final report	20	1
Final story	5	1
Total	180	

Table 9: Project's codelines.

Language	JAVA
Code lines	6806
Classes	103
Method lines of code	3625

3 Mixel

3.1 Overview

The goal of the Mixel project was to develop an XML teaching data conversion tool for Finnish universities. The final product consists of two parts: a conversion tool with a conceptual document, and a usability survey.

The conversion tool transforms XML documents from the M standard format to documents, which follow the Finnish Open University's format. The conceptual document describes which fields can be automatically transformed, which can not be converted without additional information, and which can not be converted automatically at all. The conversion tool implements most of the transformations described in the conceptual document.

The purpose of the usability survey was to plan and search different usability requirements, user groups, and use cases of applications that handle teaching data. Based on the results found out during this survey, screenshots, design plans, and guidelines were created.

3.2 Project organisation

The client of the project was the national M project group, whose contact person was Sami Hautakangas from the University of Tampere. The M project group is a part of the Finnish Virtual University, and it consists of members from different universities. The purpose of the M group is to develop and promote the exchange of teaching data between universities.

The managers of the Mixel project:

- Riina Pakarinen
- Vilppu Tuominen

The members of the project group were:

- Heini Anttila
- Pia Järvinen-Hiekkänen
- Ville-Veikko Kalkkila
- Olli Savolainen (from the usability group)
- Timo Talvitie
- Tero Tapio

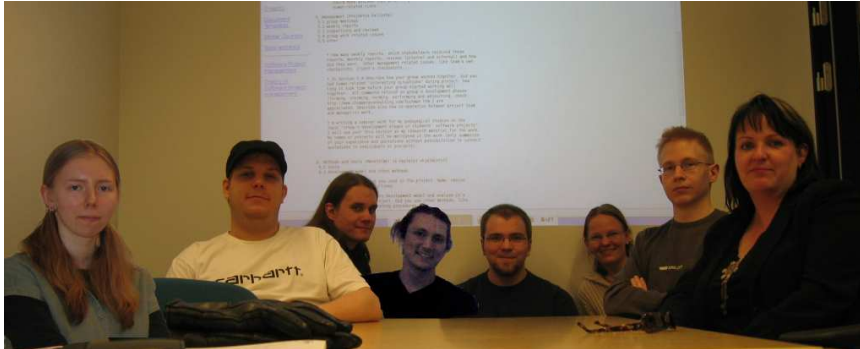


Figure 8: Project members.

3.3 Project management

During the project we arranged weekly meetings, in which we discussed about open questions, kept ourselves up-to-date about the current state of the project, checked the correctness of the timetable, and informed about coming reviews and other additional meetings. We wrote down the minutes of the meetings, sent this document to the members of the group, and archived it to the project homepage.

In addition to weekly meetings, the group communicated in the IRC and via email. The IRC was a good tool for asking advice to problems, informing about absences, or finishing pieces of the project.

During the project, we had few meetings with the client and other interest groups. Project managers were attending to these meetings, sometimes accompanied by part of or whole project group. Within these meetings we reviewed our documents, presented our project and its product, and got information about other related projects.

3.4 Methods and tools

The group used follow tools:

- Eclipse 3.2.1
- Subclipse plugin
- Java SDK/JRE 1.4.2
- Saxon-B 8.9
- XSLT 2.0

- Google Docs

We used the Eclipse editor to implement both the Java program and the XSLT transformation files. We found this useful since we were able to effectively use version control via the Subclipse plugin. We found the Saxon-B XML parser fast and easy to use, but noticed that error handling can be a bit problematic for the user program. For editing documents we used Google Docs, which was time saving as long as the documents did not include too many tables or pictures. In addition, we used Microsoft Word and Open Office for editing documents, but noticed that using two different editing programs can cause problems when using a template document.

3.5 Project phases

Our project employed an incremental development model. We chose this method because the goals of the project were initially easy to separate from each other (later on the requirements changed). The original project increments were as follows:

1. Conversion tool: from M XML to Finnish Open University XML
2. Conversion tool: from Finnish Open University XML to M XML (partially implemented)
3. Usability survey
4. XHTML conversion module for the conversion tool (not implemented)
5. PDF conversion module for the conversion tool (not implemented)

However, during the project we noticed that the first two increments could be mostly implemented at the same time (because of the underlying Java program and the similar XSLT structures). In addition, the third increment was developed alongside the first two increments. The original intention was to couple these increments within the fourth and fifth increment by using the results of the usability survey as a basis for their development, but we did not have the time to implement them. The status of the increments and individual tasks were checked regularly, and when needed, more resources were targeted to the most complicated tasks.

Table 10: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team size)
2+5+1	INC	9.10.2006	6.5.2007	209	1502,5	0,90

3.6 Conclusions

The project was challenging because of two reasons. First, it included technologies, which were new for the group. Second, some of the requirements kept changing during the project.

Our group filled some of the given goals and produced also a few additional documents, which can in the future be helpful for developers and graphical designers. We documented, among other things, ideas for future development and reasons for faced problems. The cooperation within the group worked well and the work was fairly divided.

Project areas that would've needed improvement:

- Faster reacting to types of work that took much of our time. For example, the use of an unsuitable technology or a tool, and changing requirements.
- Better feasibility study concerning selected technology.
- Clearer agreement about requirements and considering carefully before changing agreed requirements.

Project areas that were successful:

- Creating group spirit and motivation.
- Good group work and fair work dividing.
- Project group member's skills to learn and use new technology in a limited time.

3.7 Statistics

The project's working hours are shown in Figure 9. Additional project statistics are shown in Tables from 10 to 16.

Table 11: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	287,5	134,5	50	208	163	49	47	47	470,5	1456,5
%	19,7	9,2	3,4	14,3	11,2	3,4	3,2	3,2	32,3	
Usability				46						46
Total	287,5	134,5	50	254	163	49	47	47	470,5	1502,5

Table 12: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
55	29	3	5	0	0

Table 13: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence giagrams	State diagrams	Other diagrams
96	6	1	0	0	30

Table 14: Project's documents.

Document	Pages	versions
Preliminary analysis	15	2
Project plan	49	11
Requirements specification	26	3
Design plan	28	4
Test plan	37	3
Conceptual document	39	11
Usability survey	29	3
User's guide	23	1
Test report	11	4
Final report	36	2
Final story	6	1
Weekly reports	28	
Monthly reports	7	
Review reports	3	
Total	337	

Table 15: Project's codelines.

Language	Java	XML (XSLT)	XML (test material)
LOC	1454	1921	491
SLOC	546	1683	442
Reused code	0	0	0
Reused and modified code	0	0	221
Files	4	2	2

Table 16: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
9,9	390,5	0,8	12,3	7,8	41,9

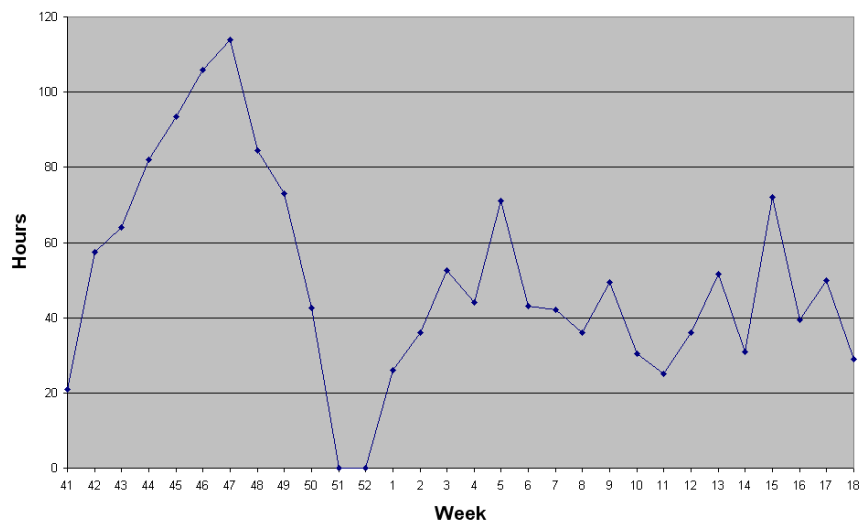


Figure 9: Weekly workload of the project.

4 SPOT

4.1 Yleiskuvaus projektista

Projektin tavoitteena oli toteuttaa www-sivusto, jonka päätarkoituksena on edistää esteettömyyden huomiointia verkkosivujen suunnittelussa ja toteutuksessa. Toteuttamamme sivusto sisältää SPOT-työkalun eli eSteettömyyden Puoliautomaattisen ja Opastavan työkalun sekä esteettömyydestä kertovan tutoriaalin. Sivusto toteutettiin Tampereen yliopiston TAUCHI-yksikölle ja se tullaan esittelemään tietojenkäsittelyopin kursseilla.

SPOT-sivusto löytyy osoitteesta: <http://www.cs.uta.fi/~spot>



Figure 10: Etusivu.

4.2 Projektioorganisaatio

Projektin asiakkaina toimivat käytettävyykslaboratorion edustaja Saira Ovaska ja aiheesta gradun kirjoittanut Leena Helin.

Projektiryhmän projektipäällikköinä toimivat:



Henri Rantala



Eerik Savolainen

Figure 11: Projektipäälliköt.

Projektiryhmäläisiä olivat:



Jenni Haataja



Juha-Pekka Honkavaara



Eise Lagerstam



Elina Saarenmaa



Joonas Tuominen

Figure 12: Ryhmäkuva.

Käytettävyys ryhmää edusti Petri Laitinen.

Joona ja Juha-Pekka tuottivat sivuston ja työkalun koodin, koska heillä oli entuudestaan kokemusta ja osaamista. Jenni, Else ja Elina tuottivat sivuston sisällön ja suunnittelivat sivuston ulkoasun ja logot, jotka Jenni toteutti suunnitelmien perusteella. Jenni toimi myöskin risk managerina.

4.3 Työskentelymetodit ja työkalut

Projektin aikana tapasimme melkein joka viikko viikkopalaverin merkeissä, jossa kävimme läpi mitä olimme saaneet aikaiseksi ja mitä seuraavaksi pitäisi tehdä. Käytimme myöskin oppimisalusta Moodlea tiedon siirtoon ja yhteyden pitämiseen.

Projektin jäsenet päättivät käyttää tekstinkäsittelyssä sellaisia ohjelmia, jotka sisälsivät mahdollisimman vähän automaattista muotoiluja, jotta tekstikapaleiden liittäminen yhdeksi dokumentiksi sujuisi mahdollisimman helposti. Lopulliset dokumentit muutettiin PDF-muotoon.

Koodi tuotettiin vapaavalintaisilla tekstieditoreilla. Spot-työkalu on koodattu pääosin ilmaisohjelmalla Crimson Editor 3.70. Projektin pääasiallisen ohjelmointikielen PHP:n versiota 5 käytettiin, koska laitoksen palvelimella, kehitysympäristössämme on kyseinen versio käytössä.

Photoshop 6.0 on ollut käytössä logojen suunnittelussa.

4.4 Projektin vaiheet

Projekti käynnistyi viikolla 42. Jaoin projektin kolmeen iteraatioon. Syksyn aikana suunnittelimme prototyypin ja toteutimme sen sekä, esituskimuraportin ja projektisuunnitelman. Ensimmäisen iteraation loppuessa joulukuun esittelimme prototyyppiä muille kurssilaisille syksyn esittelyssä. Joululoman jälkeen tammikuussa käynnistimme toisen iteraation, jonka aikana työkalu, sivusto ja sivuston sisältö saatiin siihen kuntoon, että sille voitiin tehdä käytettävyystestaus toisen iteraation lopussa. Kolmannessa iteraatioissa sivustolle tehtiin lähinnä pieniä ulkoasullisia parannuksia ja viimeistelyjä käytettävyystestin ja asiakkailta tulleen palautteen avulla.

Table 17: Yleisinformaatiota

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team size)
2+5+1	Iter.	11.10.2006	19.05.2007	221	1137,98	0,64

4.5 Yhteenveto

Kaiken kaikkiaan projekti sujui hyvin ja eteni aikataulussa. Suurempia ongelmia ei eteen tullut projektin aikana, eikä ennakoitua riskit toteutuneet. Ryhmäläiset saivat hyvän kuvan siitä mitä kaikkea projektissa mukana olemisen tarkoittaa.

4.6 Tilastoja

Table 18: Tuntimäärät työlajeittain.

Activ.	Plan	Management	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	352,42	159,58	0	81,83	152,97	0,25	18,58	0	279,85	1045,48
%	33,71	15,24	0	7,83	14,63	0,03	1,78	0	26,78	
Usab.	26	0	0	0	0	17	0	0	49,5	92,5
Total										1137,98

Table 19: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
14	10	3	10	0	0

Table 20: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
0	0	0	0	0	7

Table 21: Project's codelines.

Language	PHP, XHTML and CSS
LOC	3800
SLOC	100
Reused code	2 minor functions
Reused and modified code	0
Classes	0
Functions	6

Table 22: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
8	475	0	3	0,5	17,75

Nimi	Tunnit (h)	Osuus (%)	Käytettävyys (h)	Yhteensä (h)
Projektin johtaminen	159,58	15,24		159,58
Vaatimusten määrittely	0	0		0
Projektin suunnittelu	352,42	33,71	26	378,42
Ohjelmointi	152,97	14,63		152,97
Integrointi ja testaus	0,25	0,03	17	17,25
Katselmoinnit	18,58	1,78		18,58
Korjaus	0	0		0
Suunnittelu	81,83	7,83		81,83
Muut	279,85	26,78	49,5	329,35
Yhteensä	1045,48	100	92,5	1137,98

Figure 13: Ryhmän työmäärä aiheittain.

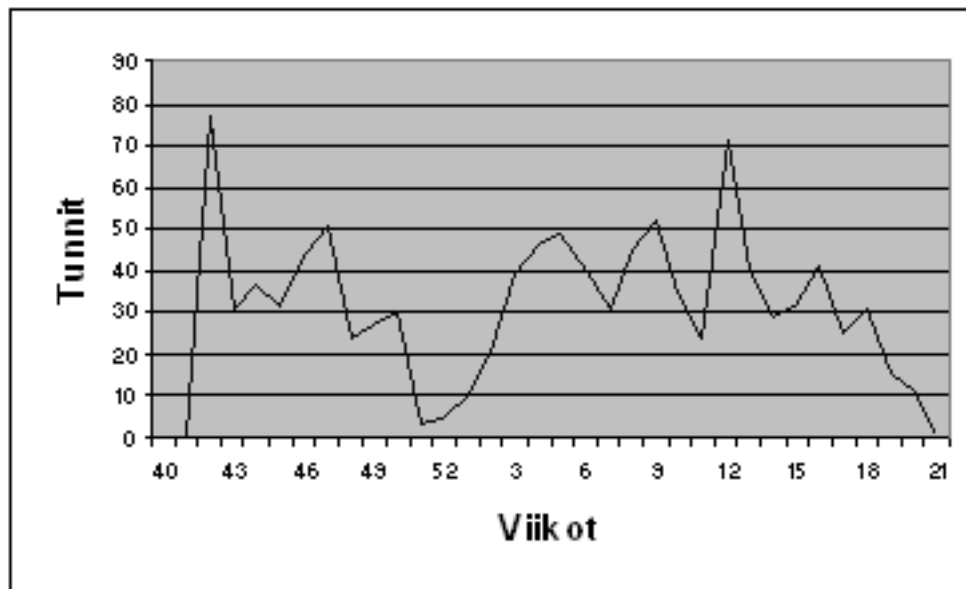


Figure 14: Viikoittaiset työtunnit.

Dokumentti	Sivuja	Versioita
Esitutkimusraportti	9	10
Projektisuunnitelma	25	16
Käytettävyyskartoitus	12	1
Käytettyvyydestin suunnitelma	4	2
Käytettyvyydestin raportti	15	4
Vaatimusmäärittely 2 iteraatio.	14	10
Sivuston 1. iteraation raportti	2	1
Työkalun 1. iteraation raportti	4	1
Ylläpitodokumentti	9	1
Loppuraportti	16	1
Projektikertomus	6	2
Viikkopalaverit	25	
Yhteensä	141	

Figure 15: Projektin dokumentit.

5 Rogue

5.1 Overview

Rogue is about a route guide application for mobile phones. The end product is called RouteGuide. See the action shot in Figure 16. Multimodality of the application supports all types of end users. They can have a vision ranging from normal to visually impaired. Multimodality is realized by speech and graphical user interfaces. System not only produces speech by speech synthesizer but it also recognizes speech input.



Figure 16: RouteGuide client side application on a mobile phone.

Graphical interfaces is designed for visually impaired and it is not traditional mobile application interface. We also implemented server component, which does the speech recognition and retrieves route information from external route guides like YTV (short haul) and Matka.fi (long distance). One of the objectives was to produce extensive UI design documents and at least paper prototype them. The UI design and implementation phase was challenging for such a different and vast target group of end users and also because of the unfamiliarity with mobile programming. Rogue web site: <http://www.cs.uta.fi/~rogue/>.

5.2 Organisation and management

Our project team consisted of two project managers: Alekski Melto and Johannes Niemelä. Group members were Ulla-Kaija Ahola (UI design), Mari Pesonen (specifying & testing), Jouni Salo (programming & architecture) and Simo Tenhunen (system & data management). Pictures of the project members are in Figure 17. We also got invaluable assistance from UTeam and especially from Eveliina Houni. Development model of the Rogue was incremental. Although the start of the project went pretty much as basic water fall model because the course forces it that way with tight documentation deadlines in the beginning. At the beginning of the year 2007, we finally got the two increments going. We held regular weekly meetings but schedule changes of the members at the middle of the spring made it impossible to find a shared weekly meeting time. In some weeks we held several meetings. At the weekly meetings we gave tasks to members and followed how the former tasks were completed.

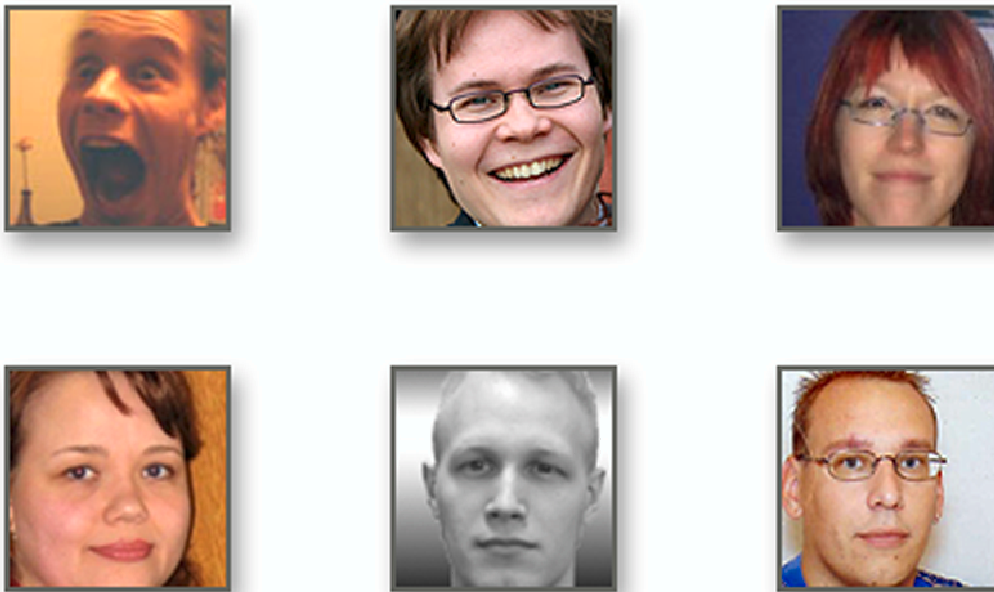


Figure 17: Rogue project members.

5.3 Methods and tools

Following list contains the most used and needed tools.

- Mobile phones for testing: Nokia 6600 and Nokia N72. We used also the members' mobiles like Nokia E70 for example. Phones were invaluable and we have hoped to get them sooner.
- Adobe Photoshop, PaintShop Pro 8, GIMP and Microsoft Visio were used with diagram, chart and image manipulations. They were useful with image manipulation and with sketching UI scenes.
- NetBeans 5.0 and 5.5 + Mobile Pack. IDE for Java ME and client programming and debugging. Very useful.
- Wireless Toolkit 2.5. Emulators for emulating client side code. Emulators are useful but they can also be tricky. It is because the differences between the emulators and actual mobile devices.
- Symbian Series 60 3rd ed. FP1 platform for Java developers. Useful for emulating.
- Java SDK 1.4.2 palvelinkomponenttien ohjelmoimiseen. Välttämätön.
- ConText text editor for modifying the code. Server programmers choice.
- Server uses the SoX 13.0.0 for converting the sound data from format to another.
- CVS version controlling. We tried to use the version controlling for our code base. At the end, it was used rarely. We think that it was because we practically had only two developers and they were doing clearly separate parts. Connections with the CVS was handled with SmartCVS 5.1.
- MS Office and OpenOffice word processors for producing the documents, spreadsheets were used for workhour tables. Presentation softwares were used for making the presentations. They were all useful but we had problems with compatibility issues between the two although we decided to mainly use freely available OpenOffice at the beginning of the project.
- Jaspis-Framework. Framework for the speech recognizer and server side as the whole.
- Zoho Projects is a web based project management tool and it was used for allocating task, using shared calendar and for storing the documents. Zoho Sheet and later EditGrid web based spreadsheets. These were really useful for tracking the hours spent with the project. Nevertheless, we had to abandon the Zoho Sheet because it did not work well and robust enough for all the project members. EditGrid worked smoothly. With these web based spreadsheets we saved weekly labour. Everybody was also aware of their personal hours but also other persons hours as well.



Figure 18: Advanced text input screen from the RouteGuide application.

5.4 Project phases

We decided to take an incremental model for Rogue's development model. We did not want to use the traditional waterfall model. We think that WF model is illfitted for this kind of mobile project which explores the new grounds of UI and speech input and output. The fall semester advances though much like basic WF project because the documentation deadlines leaves a little room for a different kind of approach. We ended up to do two increments, albeit we had planned to do the four of them. First increment was server side with speech recognition capabilities and the second was the client side with non-traditional UI and advanced address input system. Client side RouteGuide screenshot in Figure 18.

5.5 Conclusions

New kind of multimodal interfaces are very challenging to design and develop with relatively inexperienced group. Target platform, mobile phones, does not make it any easier. Instead, it raises the bar even more. We did not succeed to deliver as high quality and as feature rich end product that we would like to. But we did extensive user interface design and succeed excellently with learning the project work. Besides the difficulties with time schedules of the members, there were fine individual achievements during the project. These rush periods really pushed the project ahead. Describe here your experiences from the project and what you can do better next time! Share your experiences with other groups and readers of this document.

Table 23: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team size)
2+4+1	INC	09.10.2006	27.05.2007	231	1566,25	0,97

Table 24: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	217,25	229	87	273,75	207	3,5	81	52	240,25	1390,75
%	15,6	16,5	6,3	19,7	14,9	0,2	5,8	3,7	17,3	100%

5.6 Statistics

Weekly working hours statistics can be seen in Figure 19. Figure 19 shows workload by week but also divided into categories. Table 23 summaries general information about the project. Table 24 tells composition of working hours. Amount of documents, their page and version counts are in Table 27. Finally, Table 29 shows some productivity metrics by project months.

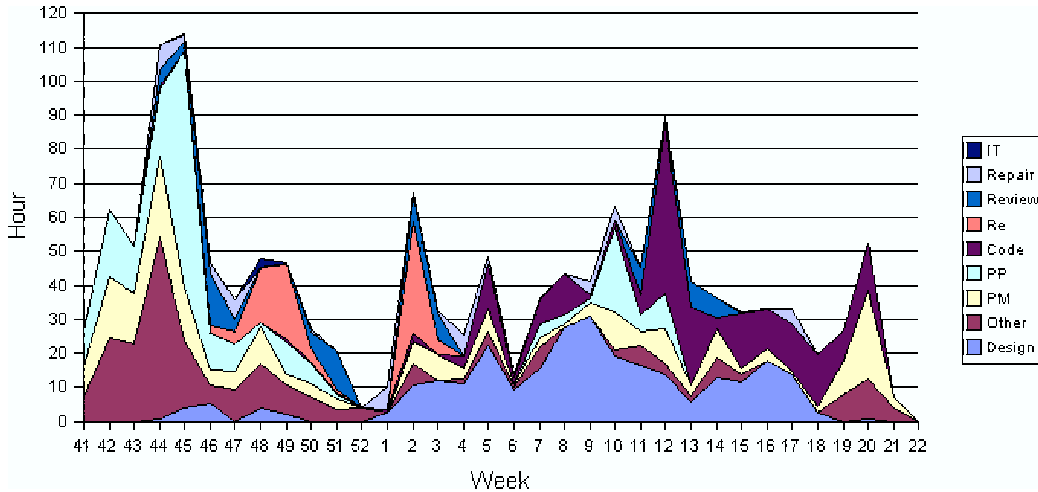


Figure 19: Weekly workload of the project by categories.

Table 25: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens
?	?	?	?

Table 26: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
?	?	?	?	?	?

Table 27: Rogue's documents.

Document	Pages	Versions
Preliminary analysis	37	6
Project plan	87	8
Requirements specification	49	7
Usability mapping of the prototype	25	3
Use cases	38	11
User interface document	36	10
Paper prototype	43	6
Plan for gathering voice corpus	11	5
Implementation plan for server	16	8
Implementation plan for advanced text input	25	11
Test plan	12	9
Test report	21	10
Project report	30	10
Project story	7	4
Weekly reports	?	
Monthly reports	?	
Review reports	?	
Total	437	108

Table 28: Project's codelines.

Language	Server	Client	Total
LOC	875	2167	3042
SLOC	803	2032	2835
Reused code	0	0	0
Reused and modified code	100	0	100
Classes	4	3	7
Functions	31	54	85

Table 29: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
10,3	295,3	0,7	9,8	5,9	41,7

6 Moreeni

6.1 Yleiskuvaus ohjelmasta

Projektina oli toteuttaa pienimuotoinen verkkojulkaisujärjestelmä Tampereen yliopiston paikallisradiolle, Radio Moreenille. Järjestelmä sisältää työkalut artikkelien käsittelyyn kuten kirjoittamiseen, muokkaamiseen ja arkistointiin. Muina ominaisuuksina sovelluksessa on RSS-syötteet, podcastit ja prompterinäkömä, joka helpottaa uutisten lukemista suorassa radiolähetyksessä.

6.2 Organisation and management

Projektiorganisaatio Projektiryhmän vetäjinä toimivat

- Arto Hirsimäki
- Mikko Kuru

Projektiryhmän jäsenet vastualueineen olivat

- Antti Finni (Dokumentointi ja testaus)
- Eetu Hämäläinen (Järjestelmien ylläpito, ohjelmointi)
- Juho Saravo (Ulkoasu ja dokumentointi)
- Heikki Santasalo (Suunnittelu, ohjelmointi)
- Taika Tulonen (Käytettävyysasiantuntija)
- Hanna Tuomisto (Ohjelmointi)

Asiakkaan yhteyshenkilöinä projektissa olivat Moreenin työntekijät päätoimittaja Marko Ala-Fossi ja äänisuunnittelija Riitta-Liisa Ojanpää.

6.3 Methods and tools

Menetelmät ja välineet WWW-pohjaisen sovelluksen toiminnallisuus toteutettiin pääasiassa PHP:llä. Osa toiminnallisuuksista vaati kuitenkin JavaScriptin käyttöä. Pohjaksi näille tarvitaan tietenkin HTML:ää. Ulkoasumäärittely sovelluksen käyttöliittymään toteutettiin CSS:llä (Cascading Style Sheets).

Tietokantana käytettiin MySQL -tietokantaa, jonka käyttämistä PHP -ohjelmointikieli tukee. Sovelluksen kannalta tietokantatyypillä ei ole merkitystä, mutta ilmaisena se soveltui hyvin projektiin.

Table 30: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
2+5+1	EVO	w. 42 -06	w. 21 -07	-	733	0,7

Tietokannan hallinnassa hyödynnettiin phpMyAdmin nimistä MySQL -tietokannan hallintatyökalua. Koska ohjelma tarjoaa graafisen käyttöliittymän ja vaatii vain selaimen, nopeutti ja helpotti se tietokannan käsittelyä huomattavasti.

6.4 Project phases

Projektin vaiheet Viikoittaisia palavereita ei näin pidetty vaan projektiryhmä järjesti aina tarpeen vaatiessa yhteisiä tapaamisia. Yhteisissä palavereissa käytiin läpi projektin etenemistä ja keskusteltiin tulevista tehtävistä sekä mahdollisista ongelmakohdista.

Projektiryhmän omien tapaamisten lisäksi järjestimme asiakkaan kanssa muutamia yhteisiä tapaamisia. Näissä tapaamisissa pyrimme selvittämään, että mitä asiakas haluaa.

Projektipäälliköt kirjoittivat luennoitsijalle viikkoraportteja viikoittain ryhmän tilanteesta. Kurssin loppupuolella viikkoraporttien kirjoitusväli harveni koska tapahtumia ei juurikaan ollut.

6.5 Conclusions

Johtopäätökset Lopputuote on onnistunut, mutta projektin eteneminen ei ollut riittävän hallittua. Projektin tekninen toteutus olisi pitänyt suunnitella paljon yksityiskohtaisemmin ja paremmin. Dokumenteissa ja projektin hallinnassa on parantamisen varaa.

Lopputuotteeseen asiakas on tyytyväinen, asiakkaan toiveet saatiin esille ja lopputuote vastaa asiakkaan tarpeisiin. Lopputuote sisältää kaikki asiakkaan kannalta tärkeimmät toiminnot. Näiden lisäksi lopputuotteessa on myös ominaisuuksia, jotka ilahduttavat asiakasta ja saavat Moreenin sivut houkuttelevammiksi.

6.6 Statistics

Table 31: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	155	72	86	32	141	6	12	0	229	733
%										
Usability	25	11	8	20	0	0	6	0	0	70
Total	180	83	94	52	141	6	18	0	229	803

Table 32: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
36	38	24	65	4	8

Table 33: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence giagrams	State diagrams	Other diagrams
51	4	0	0	2	2

Table 34: Project's documents.

Document	Pages	versions
Preliminary analysis	8	1
Project plan	15	3
Requirements specification	36	7
Design plan	9	1
Test plan	27	2
Test report	4	1
Final report	13	7
Final story	4	1
Weekly reports	21	
Monthly reports	4	
Total	141	

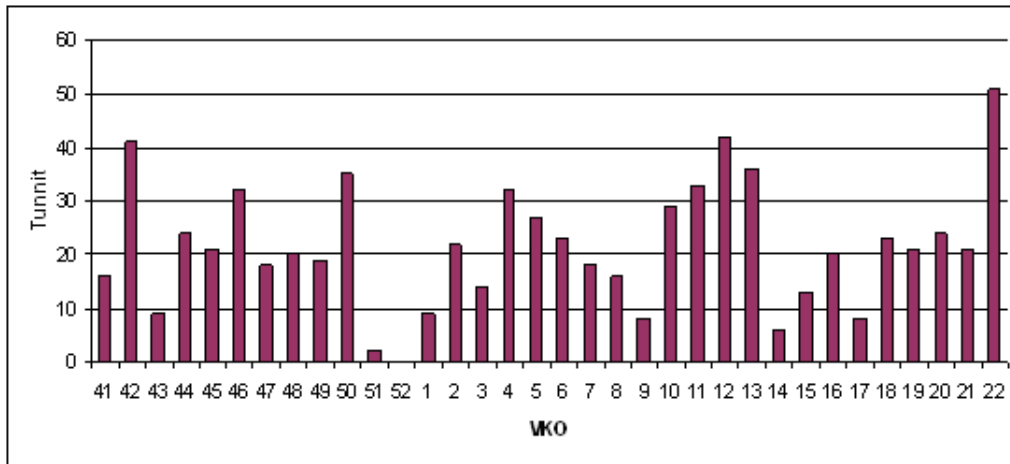


Figure 20: Projektin viikottaiset työtunnit.

Table 35: Project's codelines.

Language	JAVA
LOC	42070
SLOC	5200
Reused code	32500
Reused and modified code	4370
Classes	150
Functions	

Table 36: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
4,8	8764	31,25	10,6	175	59

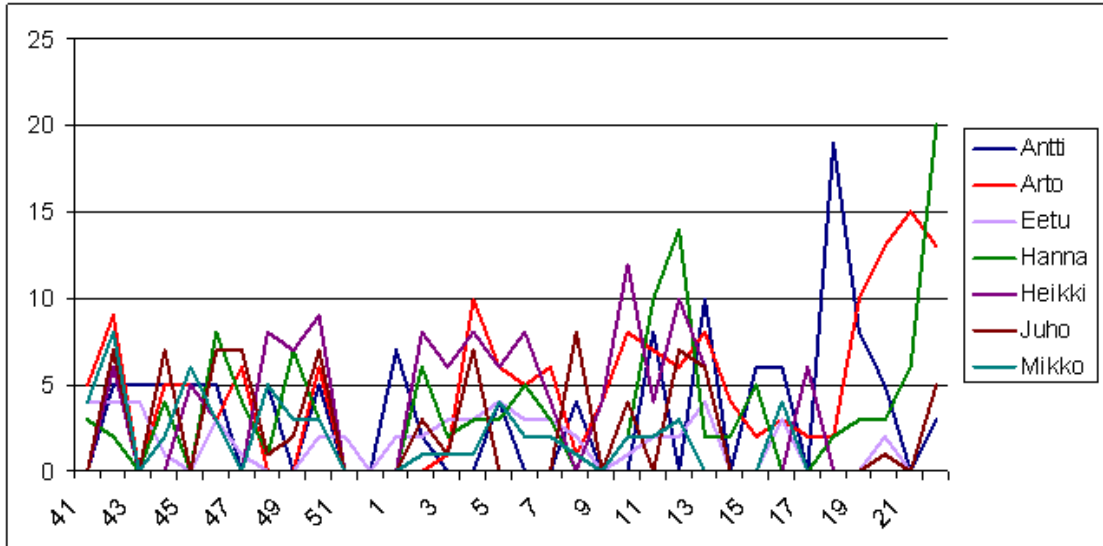



Figure 21: Henkilökohtaiset työtunnit.



moreeni

Radio Moreeni
Kalevantie 4
33014 Tampereen Yliopisto
puh 03 3551 6946
fax 03 35517250

Etusivu [Esittely](#) [Ohjelmat](#) [Uutiset](#) [Toimittajat](#) [Opetus](#) [Yhteystiedot](#) [Palaute](#)
English info

Radio Moreenin esittely

Radio Moreeni on osa Tamperetta - yhtä luonnollisesti ja varmasti kuin Pispalan harju tai Nasinneula. Moreeni on osa paikallista kulttuurielämää, yliopistoa ja tamperelaisten arkea. Moreenin maaperästä nousevat radiotyöhön rakastuneet toimittajatähdet.

Moreenin aalloilla saavat äänensä kuuluville niin merkkipäivät, julkaisut kuin tavalliset tamperelaisetkin. Me teemme ohjelmia, joita muut paikallisradiot eivät tänä lähetyvirta-aikana pysty tarjoamaan. Radiomme ääreen voi rauhoittua kuuntelemaan klassista musiikkia tai huolellisesti toimitettua puheohjelmaa. Me kerromme tarinoita ihmisten elämästä.

Moreenin runsaat ohjelmapäivät kulkevat selkeiden teemojen alla. Maanantai-illat ovat nuorison omaa ohjelmaa, tiistait kuuluvat tieteelle, keskiviikot erilaisille kansalaisohjelmille, torstait kulttuurille ja perjantai-iltana siirrymme jazzin, bluesin ja rock'n rollin kautta viikonlopuun viettoon nuorten toimittajien parissa. Lauantait ja sunnuntait ovat enimmäistä radion kuuntelu-aikaa. Silloin lähetämme järjestöohjelmien ohessa uutisoita. Mikäpä onkaan sunnuntai-iltapäivänä rentouttavampaa kuin kuunnella sivistävää luentoa tai hyvää musiikkiohjelmaa - kaikessa rauhassa.


Mikään aihe ei ole Moreenille vieras tai vähäpätinen. Tartumme rohkeasti aiheeseen kuin aiheeseen. Yliopiston tutkimuksesta on mahdollista kertoa selkein, ymmärrettavin sanoin. Tai ajren pienistä tapahtumista niin, että tilaa annetaan paitsi asioille myös tunteille. Moreenissa voi ottaa kansalaispuheenvuoroissa rohkeasti kantaa yhteisiin asioihin. Puhumalla voi myös vaikuttaa.

Moreeni viihdyttää kuulijoitaan niin uutuuslevyillä kuin vanhoilla klassikoilla. Voipa meillä kuunnella musiikkia, jota kenties muualla ette kuule. Asiantuntevat musiikitoimittajamme kertovat taustat ja tarinat, joista musiikki on syntynyt tai synnyttänyt.

Aika usein radiossa ovat paremmat kuvat kuin televisiossa. Pysykää radioaktiivisina kanavalla 98.4.

Kulttuuria Moreenista

Klassinen musiikki on aina ollut vahvasti esillä Moreenissa. Toimittaja **Siskotuulikki Toijosen** ääni



98.4 MHz
KAAPELISSA 101.5

Kuuluvuusalue 50 km Nasinneulasta. Tervetuloa samalle aaltopituudelle

BLOGI: Toimituspäällikön kynästä

TAMPEREEN YLIOPISTO

tiiedotusopin laitos

Yliopisto-TV

TREFFIPUNKTI

Figure 22: Etusivu.

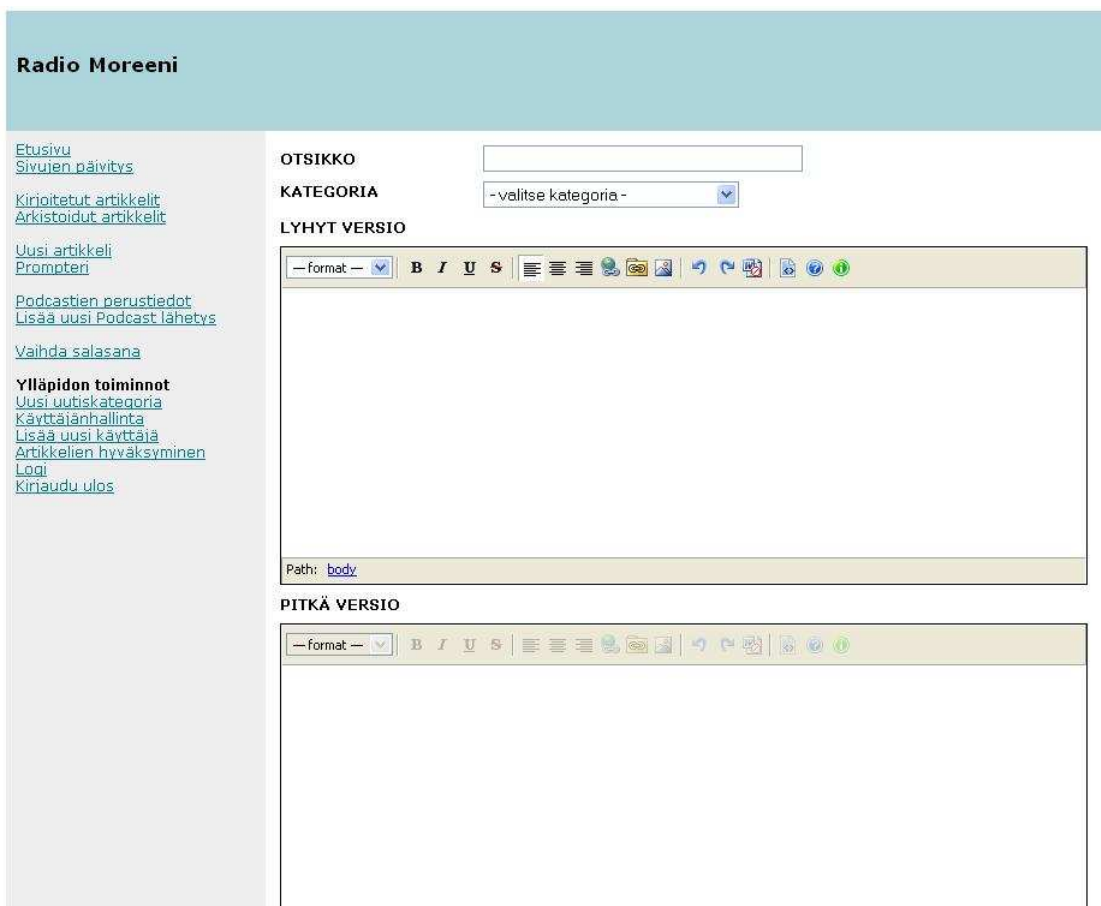


Figure 23: Admin-sivu.

7 TIKIR

7.1 Yleiskuvaus

Projektissa toteutettiin sisäinen kirjastojärjestelmä Tampereen Tietotekniikkakeskukselle. Asiakas pystyy lisäämään järjestelmään teoksia ja muokkaamaan ja poistamaan siellä jo olevia teoksia. Käyttöliittymä on www-pohjainen ja ohjelma on toteutettu LAMP-alustalle ja se tulee käyttöön asiakkaan sisäverkossa.

Järjestelmässä on kaksi tilaa. Toinen normaalille selainkäyttäjälle ja toinen admin-käyttäjälle, jolla on oikeudet lisätä, poistaa ja muokata teosten tietoja.

7.2 Projektioorganisaatio

Projektin asiakas oli Tampereen Tietotekniikkakeskus ja yhteyshenkilönä asiakkaalla toimi osastopäällikkö Marjo Soutolahti. Projektiryhmään kuului kaksi projektipäällikköä

- Pekka Peltola
- Sakari Tamminen

ja viisi jäsentä (suluissa vastualueet)

- Jussi Koro (ohjelman arkkitehtuurin suunnittelu ja sen ohjelmointi)
- Jari Porola (ohjelman arkkitehtuurin suunnittelu ja sen ohjelmointi)
- Mikko Sahanen (käyttöliittymän suunnittelu ja sen ohjelmointi)
- Reetta Övermark (käyttöliittymän suunnittelu ja ohjelman testaus)
- Petra Vyyryläinen (käyttöliittymän suunnittelu ja tietokanta)

sekä käytettävyysryhmäläinen, joka auttoi käyttöliittymän suunnittelussa ja testauksessa

- Jenny Sihvo

7.3 Projektin eteneminen

Projekti pysyi hyvin aikataulussa. Taulukossa 37 on arvioidut ja toteutuneet aikataulut.

Table 37: Aikataulut.

vaiheet	arvioidut		toteutuneet	
	alkaa	päätyy	alkoi	päätyi
esitutkimus	18.10.2006	30.10.2006	18.10.2006	30.10.2006
määrittely	15.11.2006	10.12.2006	15.11.2006	07.12.2006
suunnittelu	25.10.2006	31.03.2006	25.10.2006	14.02.2006
toteutus	15.11.2007	15.4.2007	15.11.2007	01.05.2007
testaus	10.01.2007	07.05.2007	19.03.2007	07.05.2007
käyttöönotto	07.05.2007	18.05.2007	07.05.2007	18.05.2007

7.4 Projektin hallinta

Projektiryhmä kokoontui palaveriin viikoittain, taukoa palavereista pidettiin vain kahtena viikkona. Ryhmäläiset lähettivät sunnuntaisin viikkotunnit projektipäälliköille, jotka kokosivat viikkoraportit.

Projektiryhmä katselmoi kaikki dokumentit ensin itse. Asiakas osallistui projektisuunnitelman, vaatimusmäärittelyn ja toteutussuunnitelman katselmoointeihin, näiden lisäksi asiakkaan kanssa järjestettiin kaksi palaveria.

7.5 Koodin määrä

Taulukossa 38 on listattu koodirivien määrä.

7.6 Ruutukaappauksia järjestelmästä



Figure 24: Ruutukaappaus.

Table 38: Koodin määrä

SLOC	LOC	% of SLOC	tiedosto
SQL			
57	65	87.69	./createall.sql
57	65	87.69	Yhteensä
HTML/ PHP/CSS			
54	87	62.07	./changepasswd.php
85	111	76.58	./Config.php
162	237	68.35	./csvReader.php
55	99	55.56	./DBFW.php
4	7	57.14	./etusivu.php
88	129	68.22	./getSuggest.php
2	3	66.67	./index.php
22	36	61.11	./kirjautu.php
746	1270	58.74	./Library.php
324	488	66.39	./lisaa.php
3	5	60.00	./logout.php
7	13	53.85	./md5.php
393	581	67.64	./muokkaa.php
43	86	50.00	./ohje.php
101	161	62.73	./poista.php
27	40	67.50	./Result.php
3	6	50.00	./selaa.php
37	67	55.22	./Session.php
138	189	73.02	./tikir.php
304	432	70.37	./ui.php
239	452	52.88	./User.php
289	418	69.14	./styles.css
3126	4917	63.58	Yhteensä
JavaScript			
31	43	72.09	./common.js
117	184	63.59	./smartTextField.js
148	227	65.20	Yhteensä

Haku

Perushaku

Hae: (tyhjä = hae kaikki)

[Tarkennettu haku >>](#)

Tulokset: 31 - 40 (128)

Sivu: [<](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) [12](#) [13](#) [>](#)

	Tekijä	Nimi	Tyyppi Sijainti	Hallinta
Kaikki tiedot		Vax Rdb/VMS SQL Reference manual	Huone 466	<input type="button" value="Muokkaa"/>
Kaikki tiedot		Yleiset hankintaohjeet 1998	Toimisto 4,krs	<input type="button" value="Muokkaa"/>
Kaikki tiedot	A. Hasman (R. Haux, J. van der Lei, E. De Clercq, F. H. Roger France)	Ubiquity: Technologies for Better Health in Aging Societies	Kirjasto	<input type="button" value="Muokkaa"/>
Kaikki tiedot	AAA aaaaa	Mikon kirjat		<input type="button" value="Muokkaa"/>
Kaikki tiedot	Aale Roos	Yhteyskeskuksen toiminta	tikir	<input type="button" value="Muokkaa"/>

Figure 25: Ruutukaappaus.

Kuvissa 24, 25 ja 26 on esitetty muutamia ruutu-kaappauksia sovelluksesta.

7.7 Tilastot

Kuvissa 27, 28 ja taulukossa 39 on esitetty työtuntien tilastotietoa.

Lisää

teoksia henkilöitä luokkia tyyppejä

Lisää teoksia

Teoksen nimi: * ▼

ISBN: ▼

Tekijä: ▼

Kustantaja: ▼

Painaja: ▼

Nidenro:

Vuosi:

Sijainti: Rakennus: ▼ Huone: ▼ Henkilö: ▼

Luokka: ▼ Luokkia voit lisätä [täältä](#)

Lisätietoja:

Figure 26: Ruutukaappaus.

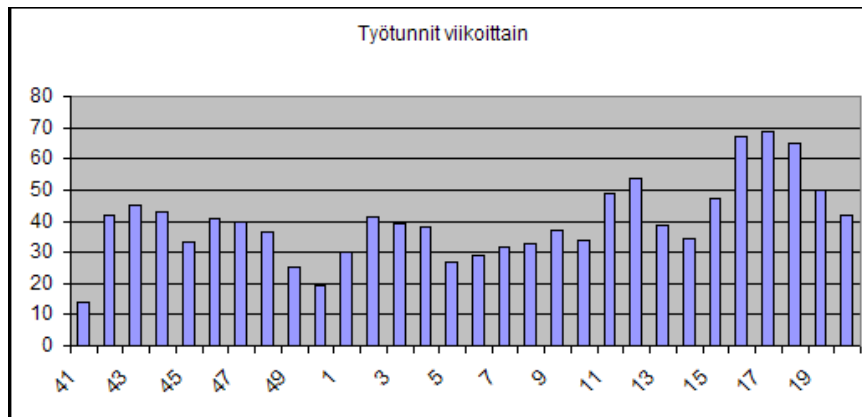


Figure 27: Työtunnit viikottain.

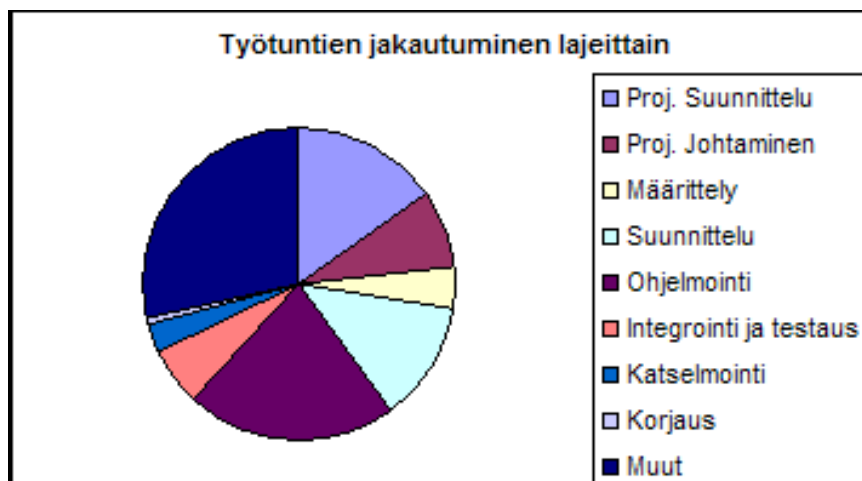


Figure 28: Työtunnit lajeittain.

Table 39: Työtunnit lajeittain.

nimi	tunnit (h)	osuus (%)
Proj. Suunnittelu	187,5	15,68
Proj. Johtaminen	96	8,03
Määrittely	50	4,18
Suunnittelu	145,5	12,17
Ohjelmointi	254	21,25
Integrointi ja testaus	75	6,27
Katselmointi	37	3,09
Korjaus	6	0,50
Muut	344,5	28,82
Yhteensä	1195,5	100

8 Sitemaps

8.1 Yleistä

Projektin tarkoituksena oli integroida Ambientia Content Manager -sovel-
luskehys toimimaan osana Google Sitemaps -palvelua. Google Sitemaps on
palvelu, jonka avulla verkkopalvelua hallinnoiva taho voi tarkastella palvelunsa
Google- näkyvyyttä. Palvelusta voi muun muassa tarkastella yleisimpiä haku-
sanoja, joilla verkkopalveluun on päädytty, sekä tilastoja Googlen indek-
soimista sivuista.

Tuotettu ohjelmistokomponentti koostaa Ambientian asiakasyrityksen verk-
kosivuista XML-muotoisen tiedoston, joka välittyy Googlen Sitemaps -palveluun.
Tiedostossa on kuvattu sivuston rakenne sekä sisältö ja metatietoa liittyen
sivuston päivitystahtiin.

Projektin kotisivu löytyy osoitteesta : <http://sitemaps.student.ambientia.fi>.

8.2 Projektiryhmän jäsenet

Projektipäälliköinä toimivat Matias Muhonen ja Juha Heinonen. Projek-
tiryhmän muita jäseniä olivat Seppo Pääjärvi, Tuomas Hynninen, Mikko
Lehtoniemi, Timo Heikkilä ja Tomi Saarinen.

Talviloman jälkeen ryhmä jaettiin kahtia toteutus -ja käyttöliittymäti-
meihin. Totetustiimiin kuuluivat Seppo, Tuomas ja Timo. Käyttöliittymäti-
mi koostui Mikosta ja Tomista.

Seuraavassa jäsenten vastuut eriteltynä tehtävien mukaan:

Table 40: Jäsenten vastuut.

Tehtävä	Vastuussa
Projektin johtaminen	Matias Muhonen ja Juha Heinonen
Esitutkimus	Koko ryhmä
Vaatimusmäärittely	Juha Heinonen, Mikko Lehtoniemi ja Tomi Saarinen
Testaussuunnitelma	Juha Heinonen, Mikko Lehtoniemi ja Tomi Saarinen
Toteutussuunnitelma	Matias Muhonen, Seppo Pääjärvi, Timo Heikkilä ja Tuomas Hynninen
Toteutus	Seppo Pääjärvi, Timo Heikkilä ja Tuomas Hynninen
Käyttöliittymä	Juha Heinonen, Mikko Lehtoniemi ja Tomi Saarinen
White paper	Mikko Lehtoniemi
Loppuraportti ja tiivistelmä	Tomi Saarinen



Figure 29: Projektin jäsenet lopputapaamisessa asiakkaan kanssa Ravintola Näsinneulassa.

8.3 Käytetyt välineet

Järjestelmä toteutettiin Java-kielen versiolla 5. Sen lisäksi hyödynnettiin seuraavia ohjelmistoja:

- Ambientia Content Collector
- Subversion
- MySQL
- Eclipse
- JUnit
- Atlassian Confluence
- Atlassian Jira

Toteutuksessa käytettiin lukuisia komponentteja. Log4j -komponentilla hoidettiin ohjelmiston lokitustarpeet. Quartz -komponentilla ajastettiin XML -tiedoston luominen. Apache Ant -käännöstyökalulla automatisoitiin ohjelmiston kääntäminen.

Toteutuksessa hyödynnettiin yleisesti hyväksi havaittuja suunnittelumalleja, kuten DAO:a. JUnit ja mock -oliot auttoivat ohjelmiston testaamisessa.

8.4 Projektin vaiheet

Projekti aloitettiin 16.10.2007 viikkopalaverilla TKT-laitoksen projektityötilassa. Viikkopalavereja pidettiin säännöllisesti. Palaverit olivat hyödyllisiä sisäisissä katselmoinneissa ja työnjaossa. Palavereissa myös sovittiin aikataulullisia asioita ja tapaamisia.

Asiakas tavattiin ensimmäisen kerran 14.11.2006 Hämeenlinnassa. Tällöin katselmoitiin projektisuunnitelma ja tutustuttiin asiakasyritykseen. Asiakas tavattiin projektin kuluessa 4.12.2006 vaatimusmäärittelyn katselmoinnissa ja 26.1.2007 testaus- ja toteutus suunnitelmien katselmoinnissa.

Koko projektin etenemisen ajan asiakkaan määrittelemät deadline't olivat hieman edellä kurssin vastaavia. Tämä ei kuitenkaan tuottanut ongelmia ryhmälle. Lopputapaaminen asiakkaan kanssa oli Tampereella 18.4 ravintola Näsinneulassa. Kurssin puolesta dokumentit katselmoitiin Timo Porasen kanssa määriteltyjen deadlinejen mukaisesti.

Projektin kehitysmallina toimi mukautettu vesiputousmalli. Malli koostuu määrittely-, suunnittelu- ja toteutusvaiheista, joiden päätteeksi katselmoidaan vaiheeseen liittyvät tuotokset. Vesiputousmalli on hyvin dokumenttipainotteinen, ja tästä johtuen varsinainen toteutus jäi projektin loppupuolelle. Normaalisti toteutus olisi aloitettu aikaisemmin.

Toteutusmenetelmistä huomioitavaa on workshoppien käyttö. Toteutus suoritettiin pääosin workshoppeissa ja optionaalisen käyttöliittymän suunnittelu tehtiin myös workshopissa. Myös osa dokumenteista kirjoitettiin workshoppeissa.

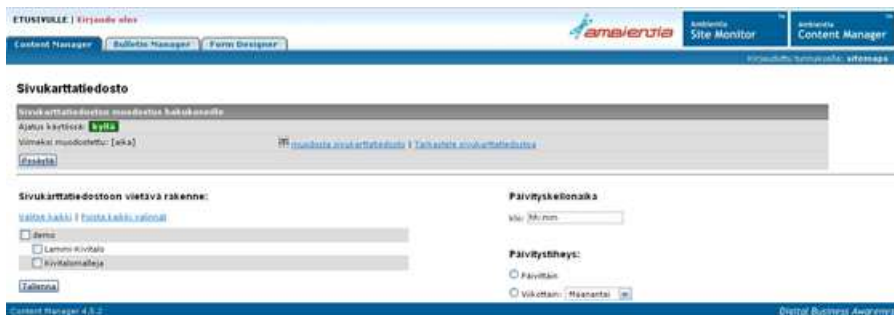


Figure 30: Käyttöliittymä.

8.5 Johtopäätökset

Projektissa ryhmä tutustui ohjelmistokehitykseen yritysympäristössä ammattimaisilla työkaluilla. Saatu käytännön kokemus koettiin arvokkaaksi. Lisäksi opeteltiin käyttämään versionhallintaa ja tehtävähallintajärjestelmää sekä testausmenetelmiä. Yhteydenpito ryhmän kesken sujui hyvin Jiran ja sähköpostilistan ansiosta. Asiakasyrityksen tuki mahdollisti projektin onnistumisen.

Ongelmia aiheutti Ambientia Content Collector -sovelluskehityksessä ilmenneet puutteet. Kehystä täydennettiin projektin aikana kattamaan ilmenneet tarpeet.

8.6 Tilastoja

Table 41: Yleisinformaatiota.

Ryhmä- koko	Keh. malli	Al. pvm.	Lop. pvm.	Päivät	Tunnit	Tunnit/ (Päivät*Ryhmä)
2+5+0	WF	16.10.2006	18.4.2007	185	626,25	0,48



Figure 31: Viikottaiset tuntimäärät.

Table 42: Tuntimäärät työlajeittain.

Työlaji	Proj. suun.	Johtaminen	Vaatim. määr.	Suunnitelu	Tot.	Katselm.	Korjaukset	Muut	Yht.
Tunnit	33,25	62,5	25,0	36,5	123,5	60,75	2	292,75	626,25
%	5,2	9,8	3,9	5,7	19,4	9,5	0,3	46,0	100,0

Table 43: Vaatimukset.

Sivuja	Vaatimuksia	Käyttötapauksia	Kayttöliittymänäkymiä
6	8	1	1

Table 44: Projektin dokumentit.

Dokumentti	Sivuja	Versioita
Esitutkimus	19	1
Projektisuunnitelma	28	3
Vaatimusmäärittely	16	2
Toteutussuunnitelma	13	2
Testaussuunnitelma	20	2
White paper	10	2
Loppuraportti	21	2
Loppuraportin tiivistelmä	5	2
Viikkoraportit	26	
Kuukausiraportit	7	
Katselmointiraportit	0	
Yhteensä	165	

Table 45: Projektin koodirivit. Uudelleen käytettyyn koodiin sisältyy Content Collectorin koodi.

Kieli	JAVA
LOC	2020
SLOC	1063
Uudelleen käytettyä koodia	69830
Luokkia	16
Funktioita	73

Table 46: Tuottavuusmetriikat suhteutettuna projektikuukausiin.

PK	LOC / PK	Luokat / PK	VM & TS sivut / PK	LOC sivut / PK	Sivut yht. / PK
4,12	490,0	3,9	7,0	9,8	49,9

9 Modulo

9.1 Yleistä

Projektin tavoitteena oli toteuttaa Tampereen yliopiston ylioppilaskunnalle erilaisia verkkopalveluita Joomla! -julkaisujärjestelmään. Vastaavia palveluita on tähän asti tarjottu perinteisin menetelmin, joten tietojärjestelmä helpottaa ylioppilaskunnan työntekijöiden työtaakkaa toimintojen siirtyessä paperilta verkkosovelluksiin.

Rajallisten resurssien vuoksi aloitimme ensimmäiseksi toteuttamaan varausjärjestelmää, joka oli asiakkaan toivoma olennaisin ja tärkein moduuli. Mikäli aikaa olisi jäänyt, projektia olisi jatkettu toteuttamalla sen jälkeen muut moduulit tärkeysjärjestyksessä. Nyt seuraavista moduuleista ehdittiin tehdä vain vaatimusmäärittelyt. Varausjärjestelmä toteutettiin asiakkaan kanssa laaditun vaatimusmäärittelyn pohjalta. Koska projekti koostui useista erillisistä työtehtävistä ja -vaiheista, käytimme inkrementaalista kehitysmallia projektin läpiviemiseksi.

Asiakkaan osalta lopputuotteen tavoitteena oli keventää sihteeristön työtä ja kasvattaa palveluiden käyttöastetta niiden näkyvyyden lisääntyessä. Projektissa käytettiin avoimen lähdekoodin ohjelmistoja, joten kustannuksia ei niiltä osin tullut.

Tamyn moduuleiden kehitystyö jatkuu todennäköisesti myöhemmin tämän projektiryhmän ulkopuolisella kokoonpanolla. Dokumentoinnissa on pyritty siihen, että ominaisuuksien lisääminen olemassa olevaan moduuliin ja työn jatkaminen olisi mahdollisimman helppoa. Myös uusien moduuleiden toteuttamista on helpotettu tekemällä niistä vaatimusmäärittelyt huolellisesti valmiiksi.

9.2 Projektiorganisaatio

Projektiryhmä koostui seuraavista henkilöistä vastuualueineen:

Projektipäälliköt:

- Marko Hassinen
- Janne Siltanen

Jäsenet:

- Sauli Alho - riskienhallinta, testaus



Figure 32: Modulo-projektiryhmä

- Mika Ekman - ohjelmointi
- Johanna Hiden - vaatimusmäärittely, dokumentointi
- Pasi Paunu - ohjelmointi
- Kalle Laukkonen - käytettävyys

Asiakas:

Projektin asiakas oli Tampereen yliopiston ylioppilaskunta (Tamy). Tamy on Tampereen yliopiston opiskelijoiden etu- ja palvelujärjestö, joka tuottaa palveluita yli 12000 jäsenelleen.

Asiakkaan yhteyshenkilö:

TAMY - Kulttuuri- ja tiedotussihteeri
Sami Kainulainen
kulttuurisihteeri.tamy@uta.fi

Joomla!-julkaisujärjestelmään siirtymisestä vastasi asiakasorganisaatiossa projektisihteeri Mikko Mikkola, mikkola.mikko@gmail.com.

Projektin ohjaamisesta kurssin puitteissa vastasi FT Timo Poranen. Käytettävyyssasioissa projektiryhmämme jäsen Kalle Laukkonen vastasi yhteydestä projektityökurssin käytettävyyksiin suuntaan.

Sidosryhmät:

Ulkopuolisenä sidosryhmänä toimi Tampereen yliopiston tietokonekeskus. Tietokonekeskus tarjosi projektin käyttöön autentikaatiopalvelut riviopiskelijoiden tunnistamiseksi järjestelmässä peruspalvelutunnuksen avulla.

9.3 Käytetyt välineet

Ohjelmointiin käytettiin työvälineenä *Crimson editor v 3.70 -editoria* sen monipuolisuuden ja konfiguroitavuuden vuoksi.

Versionhallintaan valittiin *Subversion 1.4.2*, joka mahdollisti koodin tutkimisen ja lataamisen myös ilman versionhallintaohjelmaa www-sivun käyttöliittymän kautta. Versionhallintatyökaluna oli helppo ja monipuolinen *TortoiseSVN 1.4.1*.

Projektin käytössä oli ohjelmakoodin testaamista varten myös testipalvelin:

- FreeBSD 6.1-RELEASE
- Apache 2.2.3 sekä 1.3.37_1
- Mysql 5.0.27
- PHP 4.4.4.1
- Perl 5.8.8

Projektin dokumentaatiot tuotettiin avoimella ja ilmaisella *OpenOffice 2.0:lla*.

9.4 Projektin vaiheet

Käytimme projektityössä inkrementaalista kehitysmallia, jossa projektia edistettiin vaiheittain. Aluksi suunnittelimme, että jokainen moduuli toteutettaisiin yhtenä projektin vaiheena. Projektin edetessä ensimmäisen moduulin toteuttaminen todettiin kuitenkin niin laajaksi työksi, että ensimmäisen inkrementin työstämiseen kului koko projektiaika.



Figure 33: Kuvaruutukaappaus varausjärjestelmästä

Projekti alkoi esitutkimuksella, jossa kartoitettiin asiakkaalta, mitä moduuleita kurssin puitteissa tulisi toteuttaa ja mitkä ovat yleisiä projektin reunaehtoja. Seuraavaksi tehtiin ensimmäisen moduulin vaatimusmäärittely, jossa kartoitettiin asiakkaan toiveet varausjärjestelmän toiminnallisuuden suhteen. Vaatimusmäärittelyiden pohjalta valittiin toteutuksen pohjaksi sopiva avoimen lähdekoodin moduuli ja tehtiin tarkempi toteutussuunnitelma.

Projekti ehti ohjelmointivaiheeseen, jota ei erinäisistä viivästyksistä johtuen ehditty saattaa loppuun projektikurssille varatun ajan puitteissa. Täten projektin loppuun aikataulutetut manuaalin kirjoittaminen sekä valmiin tuotteen asennus ja käyttöönotto jäivät toteuttamatta. Tuotetta testattiin ja testaus raportoitiin niiden toimintojen osalta, jotka oli ehditty toteuttaa. Kaikki työvaiheet kirjattiin omiksi dokumenteikseen.

9.5 Johtopäätökset

Projektikurssia voidaan pitää erittäin opettavaisena kokemuksena koko projektiryhmälle. Kurssin kuluessa muodostui hyvä kokonais käsitys ohjelmistoprojektin läpiviennistä. Törmäsimme monenlaisiin ongelmiin projektin eri vaiheissa. Kurssi valmistikin erinomaisella tavalla tuleviin työelämässä kohdattaviin haasteisiin.

Ryhmän ohjelmoijat saivat paljon käytännön kokemusta ohjelmiston kehitystyöstä. Merkittävin työskentelystä opittu asia oli huolellisen suunnittelun merkitys. Vaikka ryhmämme käytti paljon aikaa projektin ideointiin ja valmisteluun, olisi suunnitelmien pitänyt olla vieläkin tarkempia, jotta kaikilta ongelmilta olisi välttytty.

Table 47: Yleistä informaatiota projektista

Koko	Keh.	Al. pvm.	Lop. pvm.	Päivät	Tunnit	Tunnit/ (Päivät*Ryhmä)
2+4+1	INC	10.10.2006	16.05.2007	217	1392	0.92

Projektin ensimmäinen inkrementti osoittautui paljon odotettua työläämmäksi toteuttaa. Projektissa ei valitettavasti ollut riittävästi ohjelmointitaitoisia ihmisiä, ja ohjelmointityö jäi vain muutaman henkilön harteille. Tästä syystä jäimme jälkeen kurssin alussa asetetusta aikataulutavoitteesta. Kurssin puolesta asetetut aikataulut osoittautuivat liian tiukoiksi ryhmämme resursseihin nähden. Kurssin loppupuolella jouduimme toteamaan, että projekti jää kesken. Keskityimme viimeisillä viikoilla viimeistelemään ohjelmointityötä helposti jatkettavaan vaiheeseen ja tekemään mahdollisimman kattavaa dokumentaatiota projektia mahdollisesti jatkavalle taholle.

9.6 Tilastoja

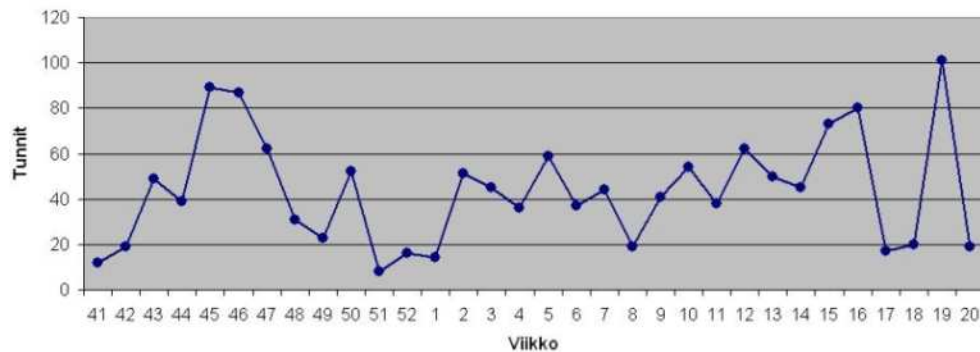


Figure 34: Projektiryhmän viikoittainen työtuntimäärä

Table 48: Työtuntimäärät aiheittain

Työlaji	Proj. suun.	Johta- minen	Vaetim. määr.	Suun- nit- telu	Tot.	Int. ja test.	Kats- selm.	Kor- jauk- set	Muut	Yht.
Tunnit	216	118	80	349	373	8	29	0	116	1289
%	15	8	6	32	27	1	2	0	9	
Käyt. YHT.				103						103 1392

Table 49: Projektin dokumentit

Dokumentti	Versioita	Sivumäärä
Esitutkimus	2	13
Projektisuunnitelma	12	37
Vaiheistustaulukko	4	2
Riskienhallintasuunnitelma	5	5
Vaatimusmäärittely	6	34
Varausjärjestelmän käytettävyysskartoitus	3	9
Muut käytettävyysskartoitus	1	9
Muut käyttöliittymämäärittely	1	8
Toteutussuunnitelma	5	14
Käyttöliittymäsuunnitelma	3	12
Testaussuunnitelma	5	16
Testausraportti	2	8
Loppuraportti	5	21
Projektikertomus	3	6
Viikkoraportit		26
Kuukausiraportit		7
Katselmointiraportit		0
YHTEENSÄ		227

Table 50: Projektin koodirivien määrä

Kieli	PHP
LOC	2074
SLOC	1183
Uudelleenkäytettyä koodia	n. 15000
Uudelleenkäytettyä ja muokattua koodia	n. 900
Luokat	6
Funktiot	34

Table 51: Tuottavuusmetriikat suhteessa projektikuukausiin

PK	LOC / PK	Luokat / PK	VM & TS sivut / PK	LOC sivut / PK	Sivut yht. / PK
9,16	226,4	0,7	9,4	4,5	29,3

10 ToVe

10.1 Yleistä

ToVe-Projektin tarkoituksena oli tuottaa Tampereen opiskelija-asuntosäätiön TOASnet-verkolle uusi verkkorekisteri vanhan järjestelmän tilalle. TOASnet-verkko on yli 6500 asukkaan käytettävissä oleva kaupunginlaajuinen alueverkko, joka on liitetty edelleen Internet-verkkoon. Rekisteri on WWW-käyttöliittymällä varustettu tietokanta, johon on koottu tiedot verkon käyttöpisteistä, IP-osoitteista, DNS-nimistä sekä verkon laitteistoista. Rekisterin tarkoituksena on mallintaa tietoverkon fyysinen ja looginen rakenne ja toimia verkon ylläpitäjien työvälineenä.

Lopputuotteen oli tarkoitus vähentää virhetilanteita, parantaa käytettävyyttä ja mallintaa entistä järjestelmää paremmin TOASnet-verkkoa. Tuotteen oli tarkoitus korvata kokonaisuudessaan vanha järjestelmä.

Projektityökurssin puitteissa projektissa toteutettiin asiakkaan vaatimusten mukainen perustoiminnallisuus, tietokanta ja vaatimusmäärittely. Kaikkia asiakkaan vaatimuksia lopullisesta toteutuksesta ei saatu toteutettua ja asiakkaan puolelta osa verkkorekisterin vaatimuksistakin on muuttunut projektin aikana, eikä siis suoraan käyttöön otettavaa järjestelmää saatu aikaan.

10.2 Projektiorganisaatio

Projektin loppuun asti ryhmässämme oli yhteensä seitsemän jäsentä, joista kaksi oli projektipäälliköitä ja yksi oli käytettävyyseriön edustaja. Yksi ryhmän alkuperäinen jäsen keskeytti kurssin.

Projektipäällikköinä toimivat

- Juha Kauppila ja
- Mikko Lammi.

Projektiryhmän jäsenet olivat

- Kati Alha,
- Sirkka Eloranta,
- Päivi Nätyynki,
- Mikko Rantanen ja

- Taika Tulonen,

joista käytettävyyssryhmää edusti Taika Tulonen.

Mikko Lammi ja Juha Kauppila olivat vastuussa projektin johtamisesta. Mikko Lammi toimi myös asiakkaan edustajana ja välitti ryhmällemme asiakkaan vaatimukset verkkorekisteristä.

Kaikki ryhmän jäsenet olivat vastuussa dokumenteista, joista pääosan teki Kati Alha. Mikko Rantanen toteutti sovelluksen kehystason. Sirkka Eloranta oli vastuussa tietokannan suunnittelusta ja mallintamisesta. Kati Alha toteutti yhdessä suunnitellun toteutuksen ulkoasun. Päivi Nätylki vastasi testauksesta. Kati Alha, Sirkka Eloranta ja Mikko Rantanen olivat vastuussa sovelluksen ydintoteutuksesta.

Projektin kotisivut löytyvät osoitteesta: <http://spock.tontut.fi/tove/>.

10.3 Metodit ja työkalut

Projektin kehityspalvelimena toimi WWW-palvelin TOASnetissä, jolle oli projektia varten asennettu PostgreSQL 8.1 ja PHP 5.0. Tietokannanhallintajärjestelmänä PostgreSQL oli tutuin ryhmän jäsenille. Ohjelmointikielistä PHP oli varteenotettavin vaihtoehto WWW-ohjelmointiin soveltuvuudensa vuoksi ja siksi, että se oli tuttu useimmille projektin jäsenille.

Projektin dokumentit tuotettiin Microsoft Wordilla.

10.4 Projektin vaiheet

Ryhmä aloitti tapaamiset lokakuussa ja tapasi lähes viikoittain yliopistolla. Projektilla oli myös oma irkkikanava, jolla suunniteltiin ja keskusteltiin aktiivisesti.

Vuoden 2006 puolella, eli projektin alkuvaiheessa, projekti eteni vesiputousmallin mukaisesti, mutta tammikuussa alkaneessa toteutusvaiheessa käytettiin inkrementaalista kehitysmallia.

Vuoden 2006 puolella tehtiin esitutkimus, projektisuunnitelma ja vaatimusmäärittely, joita ei muokattu inkrementaaliseen malliin siirtäessä enää merkittävästi.

Projektin toteutusvaiheessa käytettiin inkrementaalista kehitysmallia, jossa toteutettiin kaksi inkrementtiä. Ensimmäisessä inkrementissä toteutettiin tietokanta, kirjautumis- ja kehystaso sekä sovelluksen ulkoasu. Toisessa inkrementissä toteutettiin asiakkaan priorisoinnin perusteella perustoiminnallisuus.

Toteutettavia vaatimuksia jouduttiin karsimaan merkittävästi projektin edetessä oletettua hitaammin. Kattava käytettävyydestä jätettiin testattavan toteutuksen liian myöhäisen valmistumisen vuoksi kokonaan pois. Huhtikuussa katsottiin, mitkä vaatimuksista ovat vielä realistisesti arvioiden toteutettavissa ja asetettiin nämä projektin tavoitteeksi. Nämä saatiinkin projektin puitteissa vielä valmiiksi. Kuvassa 35 on näytönkuva toteutuksesta.

Toukokuussa keskityttiin testaamaan jo toteutettua toiminnallisuutta.

The screenshot shows a web application interface with a purple header. The main content area displays information for a property named 'Reinola'. The information includes the area 'Peltolampi', coordinates, address 'Ketunleivänkatu 4', and creation/modification dates. Below this is a table of buildings.

Rakennukset					
Id	Nimi	Osoite	Kuvaus	Asuntokohde	Rakennusryhmä
1	A	Ketunleivänkatu 4 A		Reinola	TOAS
2	B	Ketunleivänkatu 4 C		Reinola	TOAS
3	C	Ketunleivänkatu 4 D		Reinola	TOAS
4	D	Ketunleivänkatu 4 E		Reinola	TOAS

Figure 35: Kuvankaappaus sovelluksesta: asuntokohteen tietojen näyttäminen.

10.5 Johtopäätökset

Vaikka yliopiston projektityökurssi onkin monella tapaa erilainen kuin työelämän projektit, antoi se kuitenkin osviittaa työelämän projekteihin. Ryhmän jäsenet saivat hyvän kuvan ohjelmistoprojektin vaiheista ja harjoittelivat projektityöskentelyä.

Kurssin aikana havaittiin, että projektipäälliköiden rooli on erittäin tärkeä ja että ryhmätyötaitojen merkitys korostuu projektin etenemisessä ja onnistumisessa. Workshopit olivat motivoiva ja projektia paremmin eteenpäin vievä metodi kuin ryhmäläisten itsenäinen työskentely kotona. Workshopit pakottivat ryhmän jäsenet kommunikoimaan ja muuta ryhmää oli helpompi lähestyä ongelmatilanteissa kuin sähköpostin tai IRCin välityksellä.

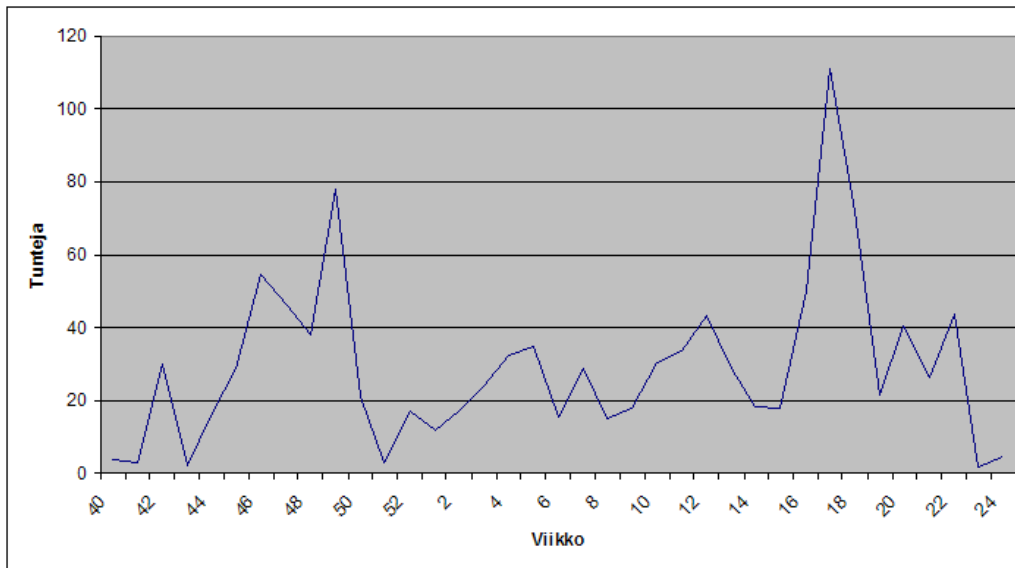


Figure 36: Projektin viikoittainen työtuntijakauma.

Table 52: Yleistä tietoa projektista

Ryhmä- koko	Keh. malli	Al. pvm.	Lop. pvm.	Päivät	Tunnit	Tunnit/ (Päivät*Ryhmä)
2+4+1	WF+INC	16.10.2006	08.06.2007	236	1087,5	0,58

Vaatimusten tarkkuuden tärkeys huomattiin erityisesti silloin, kun toteutusta oltiin viety eteenpäin suuntaan, joka ei ollut asiakkaan vaatimusten mukainen. Riittävä ohjaus kaikissa projektin osa-alueissa ei ole vähäteltävissä. Oma yrittäminen ja uusien asioiden opettelu olivat myös jatkuvasti tärkeässä roolissa projektin etenemisen kannalta.

10.6 Tilastoja

Kuvassa 36 on projektin työtuntien jakautuminen.

Taulukosta 4 huomataan, että meillä ei ollut juuri ollenkaan suunnittelukaavioita, pärjäsimme ilmeisesti, mutta niiden käyttö olisi saattanut selvittää kuvaa projektista.

Table 53: Työtuntimäärät kategorioittain

Työlaji	Proj. suun.	Johtaminen	Vaativ. määr.	Suunnitelu	Tot.	Int. ja test.	Katselm.	Korjaukset	Muut	Yht.
Tunnit	306,5	88	85,5	103	252	27	23	25	177	1087,5
%	28,2	8,1	7,8	9,5	23,2	2,5	2,1	2,3	16,3	100

Table 54: Vaatimusten ja korkean tason suunnittelun tuotokset.

Sivuja	Vaatimukset	Käyttötapaukset	Käyttöliittymäkuvat	Tietokanta-kaaviot	Tietokantataulut
61	82	-	10	1	38

Table 55: Suunnittelun tulokset.

Sivuja	Arkkitehtuurikaaviot	Luokkakaaviot	Sekvenssikaaviot	Muut
11	0	0	0	

Table 56: Projektin dokumentit

Dokumentti	Sivuja	versioita
Esitutkimus	5	2
Projektisuunnitelma	34	6
Käytettävyyskartoitus	13	6
Vaatimusmäärittely	61	14
Toteutussuunnitelma	33	8
Käyttöliittymäsuunnitelma	5	4
Testaussuunnitelma	34	7
Testausraportti	13	4
Loppuraportti	16	1
Loppukertomus	5	1
Viikkoraportit	28	15
Kuukausiraportit	4	4
Katselmointiraportit	6	4
Yhteensä	219	

Table 57: Projektin koodi.

Kieli	PHP	CSS	Javascript
LOC	7800	269	87+
SLOC	5000	200	60+
Luokat	6	0	0
Funktiot	60+	0	0

Table 58: Projektikuukausien tuottavuus.

PK	LOC / PK	Luokat / PK	VM & TS sivut / PK	LOC sivut / PK	Sivut yht. / PK
7.15	1091	8.84	$96 / 7.15 = 13.43$	21.82	$219 + 163 / 6,4 = 53.42$

11 Koodisto

11.1 Yleiskuvaus ohjelmasta

Koodisto on tietojärjestelmä, jonne on keskitetysti koottu Tullihallituksen tietojärjestelmien käyttämät lukuisat koodistot. Koodistot ovat erilaisia tietosisällön suhteen. Esimerkiksi koodisto sukupuoli voi sisältää koodin ja selitteen (M, mies). Sen sijaan koodisto valuuttakurssi voi sisältää koodin, selitteen, voimassaolon alkupäivän ja loppupäivän, määrän euroina (USD, USA:n dollari, 1.1.2006, 31.1.2006, 0,75).

Koodistojärjestelmän ylläpitoa varten toteutetaan selainpohjainen käyttöliittymä, jonka avulla koodistoja voidaan selata, lisätä ja muokata erilaisin oikeuksin. Erilaiset koodistot on voitava tallentaa järjestelmään muuttamatta tietokantaa tai käyttöliittymää. Järjestelmä tuli toteuttaa J2EE-standardin mukaisesti. Toteutuksessa tuli käyttää hyväksi Hibernate-kirjastoa sekä Struts-kehystä. Ohjelmiston perusrakenteen oli noudatettava monikerrosarkkitehtuuria.

11.2 Projektiorganisaatio

Projektin asiakas oli Tullihallituksen tietohallinto ja edustajana toimi Leena Vaarala.

Projektiryhmän projektipäällikköinä toimivat Terhi Kivinen ja Anssi Koskela. Projektipäälliköiden lisäksi ryhmään kuului alla olevat henkilöt.

- Ristomatti Airo
- Mikko Malinen
- Miika Nykvist

Lisäksi yksi projektin jäsen keskeytti marraskuussa. Käytettävyyserhmän edustajana toimi Olli Savolainen.

11.3 Projektin eteneminen

Projekti eteni alkuperäisen aikataulun mukaan työmäärien jakaantuen kuitenkin eri tavoin kuin alun perin oli arvioitu. Tapaamisiin, palavereihin ja projektin hallintaan kului enemmän aikaa mitä oli suunniteltu. Tämä johtui käytännössä siitä, että projektin aihepiiri oli vaativa ymmärtää. Projektin hallinto tekikin projektinaikana päätöksen, että jokaisen projektin jäsenen on tärkeä ymmärtää mitä olemme tekemässä ja tämän vuoksi tapaamisiin

käytettiin paljon aikaa. Samasta syystä johtuen suunnitteluun käytettiin paljon aikaa. Suunnittelu-aikaa kasvatti omalta osaltaan myös uudet tekniikat.

Toteutusvaiheessa työt oli jaettu siten että yksi toteuttajista vastasi tietokannasta toinen käyttöliittymien ulkoasun luonnista HTML sivuin ja kolmas koodasi Javalla. Vaikka tiedossa oli, että työmäärät eivät jakaudu tasaisesti, tehtävät haluttiin jakaa osallistujien kiinnostuksen mukaan. Jos toteuttajia oli ollut alkuperäinen määrä työmäärä olisi jakautunut tasaisemmin. Alla oleva kuva havainnollistaa projektin etenemistä.

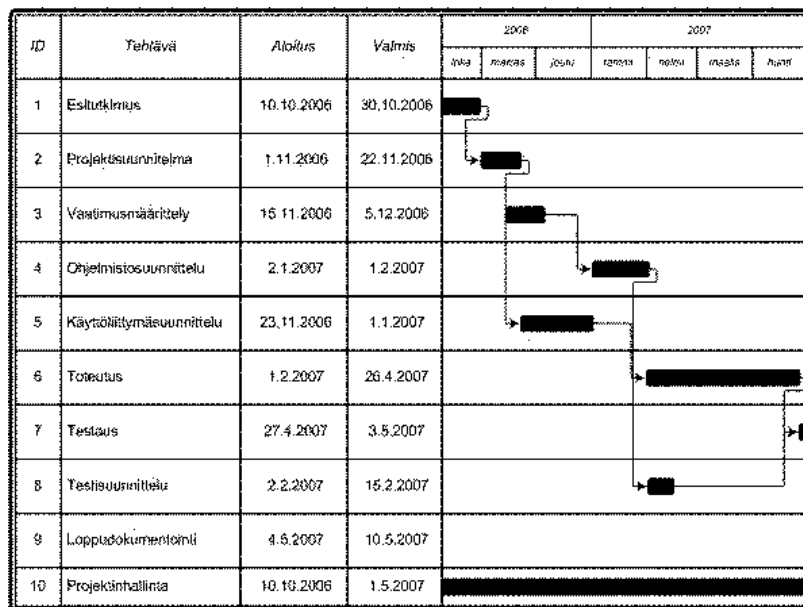


Figure 37: Projektin eteneminen.

11.4 Johtopäätökset projektista

Todelliseen, käytössä olevaan sovellusarkkitehtuuriin tutustumiseen oli mielenkiintoista. Järjestelmän monikerrosarkkitehtuuri oli haastava suunnitella ja toteuttaa. Projektityöskentely oli hyvä kokemus eri vaiheineen ja dokumenttien aikataulut ja sisällöt rytmittivät projektin hyvin. Kaiken kaikkiaan projektillaiset totesivat projektin antaneen hyvän näkemyksen todellisista projekteista ja niissä työskentelystä. Työvälineet Eclipse, Struts ja Hibernate tulivat hyvin tutuiksi. Toteuttajia olisi kaivattu enemmän: projektipääällikköjen määrä (2kpl) oli ylimitoitettu toteuttajien määrään nähden.

11.5 Projektin hallinta

Palaverit

Projektiryhmän kokoukset pidettiin pääsääntöisesti viikoittain, yleisimmin tiistaisin. Osa ryhmäläisistä oli työssä käyviä, joten kokoukset pidettiin yleensä iltaisin. Kokoukset kestivät vaihtelevasti tilanteesta riippuen puolesta tunnista useisiin tunteihin. Joululoman aikana kokouksia ei pidetty samoin pari pitempää työskentelyjaksoa hoidettiin ilman kokousta, mutta tällöinkään kokousten välillä ei ollut kahta viikkoa enempää. Asiakas osallistui suunnittelukokouksiin kaksi kertaa.

Viikkoraportit

Projektiin käytetyt tunnit ilmoitettiin projektipäällikölle viikoittain projektiseurantaa varten. Projektipäällikkö toimitti viikkoraportin ja kuukausiraportin kurssin vetäjälle viikoittain. Tammikuun lopussa raportoinnista vastaavaa projektipäällikköä vaihdettiin.

Tarkastukset ja katselmoinnit

Tärkeimmät projektin dokumentit katselmoitiin kurssin vetäjän ja asiakkaan toimesta. Projektiryhmä katselmoi myös sisäisesti dokumentit ennen niiden luovuttamista varsinaiseen katselmointiin.

Muut

Useimmat dokumentit jaettiin ryhmäläisten kesken kirjoitettaviksi. Joidenkin dokumenttien työstämiseen käytettiin wikiä, jolloin kaikki ryhmäläiset pystyivät seuraamaan toistensa osuuksien edistymistä ja tarvittaessa myös kommentoimaan niitä suoraan. Lisäksi ryhmäläiset käyttivät sähköpostia ja sinne luotua postituslistaa avuksi tiedottamisessa ja keskusteluissa.

11.6 Käytetyt välineet ja menetelmät

Välineet

Ohjelmiston toteuttamiseen käytettiin seuraavia välineitä:

- Eclipse IDE 3.2
- Java 1.5

- Web Tools Platform (WTP) plug-in for Eclipse
- Concurrent Version System version management
- PostgreSQL 7.4.7
- Tomcat
- SSH2
- phpPgAdmin 4.1
- wiki

Työvälineet helpottivat ohjelmiston toteutusta huomattavasti.

Menetelmät ja tekniikat

Ohjelmiston toteutuksessa on hyödynnetty Struts sovelluskehystä ja Hibernate-kirjastoa.

Projektimallin sopivuus projektiin

Käytetty projektimalli sopi projektiin hyvin, vaikka sitä vähän muokattiin projektin aikana aikatauluvaatimusten johdosta. Puhdasta GUIDe -mallia emme projektissa noudattaneet, vaan malli oli vesiputousmallin ja GUIDe -mallin välistä. Toteutus tehtiin iteratiivisesti toiminnallisuus kerrallaan valmiiksi. Toteutuksen aikaista projektimallia ei suunniteltu projektin alussa, mutta se otettiin käyttöön teknologian epävarmuuden vuoksi.

11.7 Tilastot

Table 59: Yleisiä tietoja projektista

Koko	Keh. malli	Al. pvm.	Lop. pvm.	Päivät	Tunnit	Tunnit/ (Päivät*Ryhmä)
2+3+1	WF/GUIDe	09.10.2006	21.05.2007	225	1046	0.77

Table 60: Työtunnit tehtävittäin

Työlaji	Proj. suun.	Johtaminen	Vaativ. määr.	Suunnittelu	Tot.	Int. ja test.	Katselm.	Korjaukset	Muut	Yht.
Tunnit	266	94	49.5	40	276	29	9.5	13	118.5	895.5
%	25	9	5	4	26	3	1	1	26	
Usability										150.5
Total										1046

Table 61: Vaatimusmäärittely ja korkean tason suunnittelun tulokset

Sivuja	Vaatimukset	Käyttötapaukset	Näytöt	Tietokantakuvaukset	Tietokantatauluja
56	28	7	16	1	11

Table 62: Suunnittelun tuloksia

Sivuja	Yleisiä kuvia	Luokka-kaavioita	Sekvenssi-kaavioita	Tila-kaavioita	Muita kaavioita
32	1	4	2	0	0

Table 63: Projektin dokumentit

Dokumentti	Versio	Sivuja
Esitutkimus	1.0	6
Projektisuunnitelma	1.1	20
Vaatimusmäärittely	1.1	25
Käytettävyysuunnitelma	1.0	10
Testaussuunnitelma	1.1	31
Toteutussuunnitelma	1.1	32
Käyttöliittymäsuunnitelma	1.1	27
Testausraportti	0.6	6
Ylläpito-ohje	1.1	26
Loppuraportti	1.0	26
Loppuraportin tiivistelmä	-	4
Yhteensä		212

12 English Usability Team

12.1 Overview

Usability team 2006-2007 consisted of students participating in the project work course. The team worked in cooperation with all the projects of the course. Each member of the team worked also as a member in a single or many other projects in the course.

The idea for a usability team came from two things: 1. The need to have better usability for each project in the yearly project work course 2. The need for students of Interactive Technology to participate in the project in ways that suit their studies. The usability team was introduced for the first time in the 2005 - 2006 project work course. It aimed at two goals: First, to provide a course fitting for the Interactive Technology students. Second to make sure the usability needs of each project in the course were properly assessed and met, possible possibly with the resources in use. The course was a pilot project, which encountered many problems, yet managed to work adequately well for the projects. And most importantly the course provided valuable information for the future courses to come. The basic idea of the team was found suitable, and it was decided to continue the project into the future.

The usability team for the 2006-2007 semesters was only the second usability team for the project work course. So some problems were predicted to be encountered. However, with the introduction of project managers and a more clear idea of the documentation, the project proceeded quite smoothly.

Note: Although providing meaningful project work course for the students in Interactive Technology is an important idea behind the usability team, it will not be referenced in other sections of this document, because this does not relate to the function of the usability team, which the remainder of the document is about.

12.2 Organisation and management

The usability team consists of 12 members and 2 projects managers. The team is split into the English and Finnish groups. The English group consulted eight English projects and the Finnish group consulted 10 Finnish projects. Each UT member was assigned to one or two projects. Some of them also helped the other projects.

UT used the distributed organisational structure where usability experts were assigned to work with separate project teams and they reported to the usability project manager. This structure allowed the usability expert

to work with the project development team. Staying with the project team from the beginning to the end of the project increased the chances of usability recommendations being implemented.

This year (2006-7) the usability project manager assigned tasks to the usability team members and held team meetings to discuss progress and issues related to working with their assigned project teams. In addition, an online Moodle forum was created, where the usability team members could discuss and exchange ideas.

Each member was assigned to work for maximum of 255 hours. Each member filled in a timesheet every week and the usability project manager approved the hours and made sure that members did not cross the limit. If a UT member had too much work then the work was split with another member with less work load in order to balance the working hours. Since each project had different requirements and some usability experts worked on more than one project, it was important to maintain a balance in the number of working hours.

12.2.1 English usability team members

- Deepa Mathew (Usability Project Manager). Email: deepa.mathew@uta.fi
- Lily Lin-Marsalo Projects: System for Multiple Language Learning (SYSMLL) and Calendar Application. Email: Lily.Lin-Marsalo@uta.fi
- Markku Myllylahti, project: USVA. Email: mm69895@uta.fi
- Minna Kause, project: OurTown. Email: Minna.Makipaa@uta.fi
- Riki Kawakami, projects: International Olympiad in Informatics (IOI) and Diachronic Internet Corpus of English (DICE). Email: Riki.Kawakami@uta.fi
- Tuomo Hakaoja, projects: Electronic Dictionary Project (DICT) and Webpage Rating and Feedback System. Email: Tuomo.Hakaoja@uta.fi

12.2.2 Group meetings

The usability team had team meetings every alternate week to discuss the issues or even to improve or suggest design improvements with the usability



Figure 38: Usability Team. From left: Tuomo Hakaoja, Markku Myllylahti, Minna Kause, Lily Lin-Marsalo and Deepa Mathew(PM).

team members. Everyone had a right to bring problems she/he had encountered during the project into discussion with other usability team members.

At the end of every team meeting the team fixed the time for the next meeting and if at anytime any member could not make it to the team meeting they had to inform the UT project manager in advance.

Before every team meeting the UT project manager sent the meeting invitation with the agenda. So that the members could think beforehand what they want to talk in the meeting. So far, we have had approximately 15 team meetings.

12.2.3 Weekly reports

The weekly reports used the template that was given in the course website. The UT project managers had to submit weekly reports to the client about the progress and the number of hours spent on the project by the team. The reports were sent by email to the client in the beginning, and then later, the weekly report was uploaded to the usability team's homepage.

12.3 Methods and tools

12.3.1 Methods

Since each project had different requirements different methods were used.

Below are some of the methods used by the usability members in general:

- Heuristic Evaluation: Usability inspection method, where the usability expert evaluates the software against a list of established heuristics or guidelines.
- Individual Interviews: One-on-one discussions with users to learn how a particular user works and enables to probe on a user's attitudes, desires and experiences.
- Personas: A fictional person that represents one of the major user groups for the site. The design team considers the needs of this person when developing the software.
- Prototyping: Draft model of the project that allows the design team to explore ideas before fully implementing them. A prototype can range from a paper prototype to interactive prototypes.
- Task Analysis: Method that involves learning about users' goals - what they want to do with the software and understanding the tasks that user will perform with it.
- Usability Testing: One-on-one sessions where a "real-life" user performs tasks with the software in order to identify user frustrations and problems with the software.
- Use Cases: Description of how users will use a particular feature of the software. Use cases provide a very detailed look at how users interact with it, including the steps a user will take to accomplish each task.

And the next list shows the how many times different methods were applied.

- Personas (2)
- Cognitive Walkthrough (3)
- Heuristic Evaluation (3)
- Paper Prototype Testing (2)

- Task Analysis (4)
- Peer Review (1)
- Usability Testing (4)
- User Interface Interviews (2)
- Use case (2)

And the next list gives the total number of document pages produced by the English language usability team.

- Usability Analysis (50 pages)
- User Interface Plan (87 pages)
- User Interface Evaluation (11 pages)
- Usability Testing Plan (4 pages)
- Usability Testing Report (34 pages)
- Paper Prototype (11 pages)
- design review report (12 pages)
- walkthrough report (38 pages)
- Use case (24 pages)

12.3.2 Tools

The most obvious usability tool for the usability team was the usability lab itself. But since each project that the usability member worked for had different requirements, not all of them benefited from the usability lab. Anyway the project that needed testing and design improvements benefited with a good, well equipped usability lab. It had a huge impact on the efficiency and effectiveness of a team and also on working with usability lab vendor to improve their products.

Another main tool used in the usability team is the tools for communication. Email has been the primary means of communication but since the team required more interactions among team members, online Moodle forums for discussions were used. Apart from that, basic tools like MS office and so on were used for documentations.

12.4 Conclusions

12.4.1 Experiences from the project

Over all the usability team work was a success. The team got to learn many things and also gained good work experience working with real time project with real clients.

The UT members are very happy with the work and are proud of it. We got to use our knowledge and experience.

Luckily the usability team did not face much group work related issues within the team but faced problems with the project team since they did not have a clear idea about the usability team's task in the project development process. Also the usability team member was working as part of the project team, but some of the project team did not consider them as a team member

12.4.2 What to do better next time

- Two usability member for each project.
- One member from TAUCHI unit to support the usability team.
- Only members who took part in the project course can become PM since they have better idea about what was done and what can be done.
- Better guide lines for usability team
- Clear job descriptions given to the project team members about usability team's task in the project development.

12.5 Statistics

Usability team members report their hours only to the usability project manager. Hour reports also dictate to which project the hours were worked for. Below is the total number of hours each UT member spent in total: (taken on 22-May-2007) Deepa Mathew (PM): total 241 hours Lily Lin-Marsalo: total 190.5 hours Markku Myllylahti: total 244 hours Minna Mäkipää: total 145 hours Riki Kawakami: total 80 hours Tuomo Hakaoja: total 237 hours

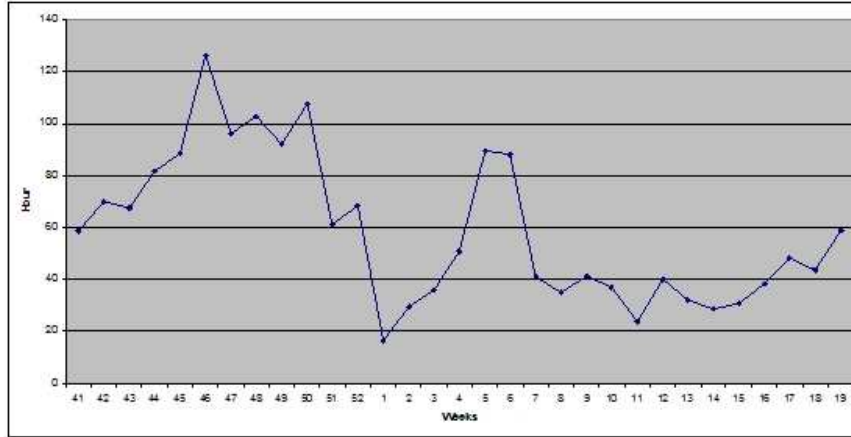


Figure 39: Weekly workload of the usability team.

Table 64: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
1+5	N/A	16.10.2006	30.05.2007	217	1137	0.88

Table 65: Usability documents.

Document	Pages	versions
Preliminary analysis	7	1
Usability plan	31	2
Final report	26	3
Final story	6	1
Weekly reports	31	
Monthly reports	4	
Total	105	

13 Electronic Dictionary Project

13.1 Overview

This project's goal was to produce a printed and online version of a dictionary of medical vocabulary in English 1375-155. The dictionary is a 6 years work of Mr. Juhani Norri, and it has special reference to names of body parts, sicknesses, instruments and medical preparations. The original Paradox database for the dictionary was created by people at the department of computer science, and our group's mission was to convert the database into PostgreSQL and create a printed version and if possible, an online version of it. However, due to the different skills among the members of the group and the resulting unequal distribution of tasks, only the transformation of the database and the creation of the printed version were accomplished.

13.2 Organisation and management

Project Managers:

Vassia Efstratiou, v_efstratiou83@yahoo.gr
Kemal Kagan Isitan, Kemal.isitan@uta.fi

Group Members:

Antti Toivanen, antti.toivanen@uta.fi
Kimmo Rinteela, kimmo.rinteela@uta.fi
Eeva Luoto, eeva.luoto@uta.fi
Petteri Valtanen, petteri.valtanen@uta.fi

Usability Team Member:

Tuomo Hakaoja

Our client:

Mr Juhani Norri, Lecturer in School of Modern Languages and Translation Studies, English Department UTA

Here is our project's homepage: <http://edp.cs.uta.fi>.

13.3 Methods and tools

In this project, we already had a database. It was created in paradox and its user interface in a free software tool called devexpress. This database was



Figure 40: Project members Petteri Valtanen, Kimmo Rinteelä, Vassia Efs-tratiou, Tuomo Hakaoja and Kemal Isitan.

to be studied but will have no further use. We decided to create a new one. The code needed was written in php 5. Tools used by our team during the project process were:

- Proposed tool for implementing the UI is php 5.0 with pear 1.4.0
- The Apache 2.2.3 HTTP server
- pxttools 0.0.20
- Internet Explorer 6.0, Mozilla Firefox 1.5
- OpenOffice.org 2.0
- Debian 3.1, kernel version 2.8.17
- Template Engine, maybe flexy or smarty (remains undecided)
- PostGreSQL 8.2 DBMS
- Adobe Photoshop
- Phpdoc, Emacs, Crimson Editor, Edit, ms word
- Turtoise.sub subversion.

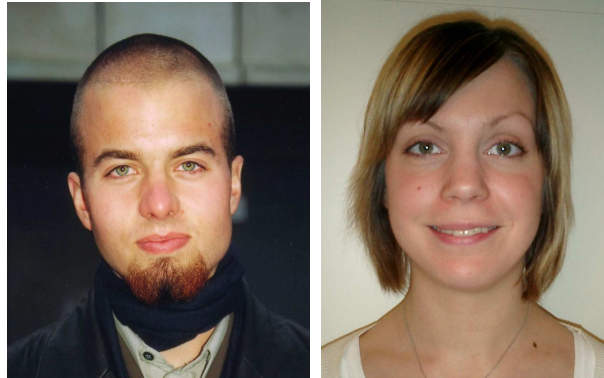


Figure 41: Project members Antti Toivanen and Eeva Luoto.

13.4 Project phases

Our first plan of the project phases was the following (source: project plan version 2.1):

The work breakdown structure defines the hierarchy of project tasks, sub tasks and work packages. Completion of one or more work packages results in completion of a subtask, and completion of subtask leads to completion of a task and furthermore to completion of the project. It is a tool to budget cost.

The electronic dictionary project is divided in three parts which one of them has its own subparts. The first one consists of project planning, requirements specifications, design of new database and user interface. The rest two begin where part one stops. Second has to do with the printed version and includes database update and user interface. Finally the third one deals with the online version its database and interface. The deadlines for each one of them is showed below.

PHASE 1: January 2007. Preliminary Analysis

SUBTASKS:

a) Project Planning October 2006

Become familiar with current database

Create preliminary project plan

Create project logo

Divide responsibilities among the group

b) Requirements Specification November 2006

SUBTASKS:

Table 66: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
2+4+1	WF	13.10.2006	31.5.2007	231	1037,5	0,64

Decide programming language
 Decide software tools
 Create requirements specification document
 Create project planning
 c) Design December 2006
 SUBTASKS:
 Design new user interface
 Design new database structure
 d) User Interface January 2007

PHASE 2: April 2007 Printed Version

- a) Client updates the new database March 2007
- b) Create user interface for printed version April 2007

PHASE 3: May 2007 Online Version

Finalize user interface for online version

In Figure 42 you can see what the printed version of the dictionary looks like for the term 'belly'. The layout of the printed version is based on the preferences of the publisher.

13.5 Conclusions

This team managed to deliver the printed version of the dictionary. Online version which was also designed although not completed otherwise, is an idea for further development of the project. It can be characterised as a successful first attempt of six people that have never worked in such a level project before.

13.6 Statistics

Table 67: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	293	101	88	127	107	32	32	12	153	945
%	31	11	9,5	14	11,5	3	3	1	16	
Usability	20			72,5						92,5
Total	313	101	88	199,5	107	32	32	12	153	1037,5

Table 68: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
24	38	0	11	2	26

Table 69: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
?	3	0	0	1	3

Table 70: Project's documents.

Document	Pages	versions
Preliminary analysis	5	1
Project plan	29	3
Requirements specification	24	2
Test plan	19	1
Implementation plan	15	1
Test report	7	1
Final report	19	1
Final story	6	2
Weekly reports	14	
Monthly reports	1	
Client meeting reports	17	
Total	156	

Table 71: Project's codelines.

Language	PHP
LOC	555
SLOC	322
Functions	13
Language	Javascript
LOC	246
SLOC	153
Functions	11
Language	(PL/PG)SQL
LOC	245
SLOC	139
Functions	13
	Total
LOC	1046
SLOC	614
Functions	37

Table 72: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
6,8	153,8	0	3,5	3,1	26,0

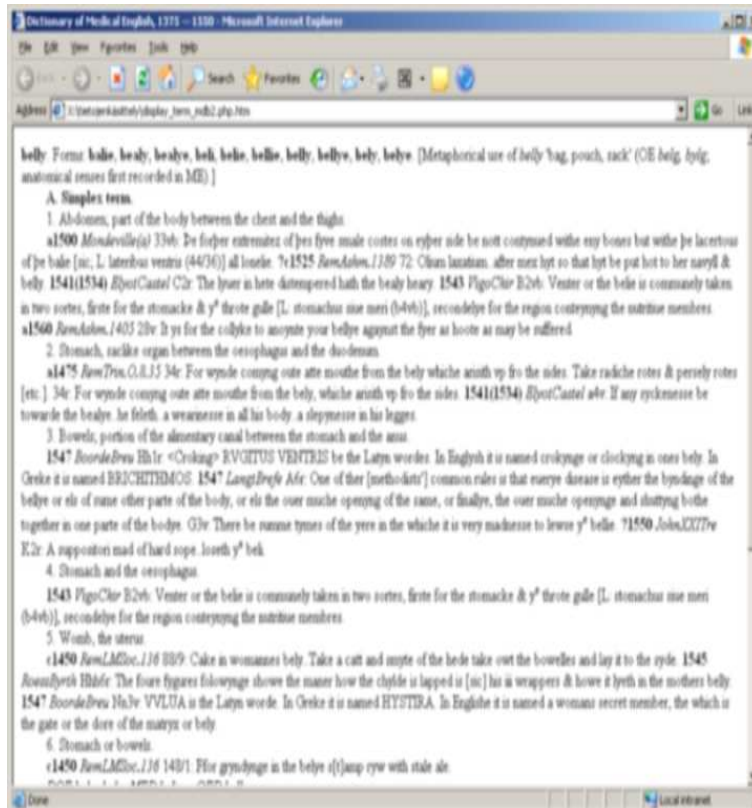


Figure 42: A demo of the printed version on term 'belly'.

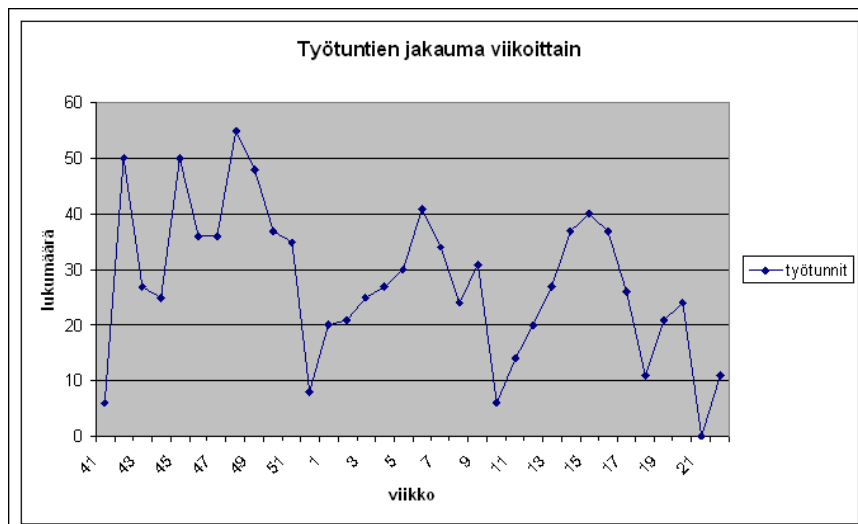


Figure 43: Weekly workload of the project.

14 OurTown

14.1 Overview

The intended product of this project is a computer software for mobile devices, based on an industry software platform. OurTown is a mobile phone application that lets people add content to different virtual locations, which represent actual real world locations. The content can be for example text, pictures and votes. Users can also modify location based information and are able to browse the content which other users have added to the system.

The system enables people to communicate in a new way. OurTown effectively creates a virtual community and a social network to connect people, and to better understand peoples' preferences and characteristics. The idea behind the program is to build a social network based on the places that people visit and like.

OurTown is basically a standalone program. To enable all the system functionalities GPS signal should be available. OurTown uses MUPE platforms resources in handling information and media files. Access to these services via Internet is required.

14.2 Organisation and management

The group is comprised of managers Umair Azfar Khan and João Carlos Andrioli Machado; usability expert Minna Kause; and developers Simo Fält, Teppo Tomann, and Pellervo Valkama. Two project members quit the course.

The management of the project followed waterfall software development model for the first 5 months; after that, it was shifted to incremental model. The main reason for this change was to extreme lack of information about what the client wanted as a final product and also about the framework on which the project was relying (Mupe). The managerial work was divided between the Umair and Carlos after November. Umair finished his role in the end of January and Carlos took over until the end of the semester. The other members were lead with no discrepancy about their main functions i.e. all members were developers. The management style differed during the year, as Umair was very clear when defining the activities that each member was supposed to execute. Carlos gave more freedom and independence to the group members.

The group was reasonably easy to manage since everybody was eager to help and cooperate. However, the project execution suffered severely from customer's unavailability, lack of adequate equipment and infrastructure; and immaturity of the Mupe platform. These problems are the main cause for

the project not reaching its end with a complete application as a product.

The group had weekly meetings fairly regularly until the end of February. After that it seemed that the meetings were held at longer intervals. This was unfortunate as it seemed that the project was not progressing as planned. This coincides with the period of time where the group was expecting to have the necessary infrastructure and equipment promised by the client, but for some unknown reason was delayed.

The group reported working hours to the managers via project's website <http://vortex.fi/ourtown> and also via Excel XLS files. The website was by far a better choice since it makes the reporting much simpler for all members. Also because some Excel XLS reports were lost due to system crashes on the machines they were being hosted.



Figure 44: Project members - Teppo, Pellervo, Carlos, Umair, Simo.

The figure 45 shows a screenshot of the system.

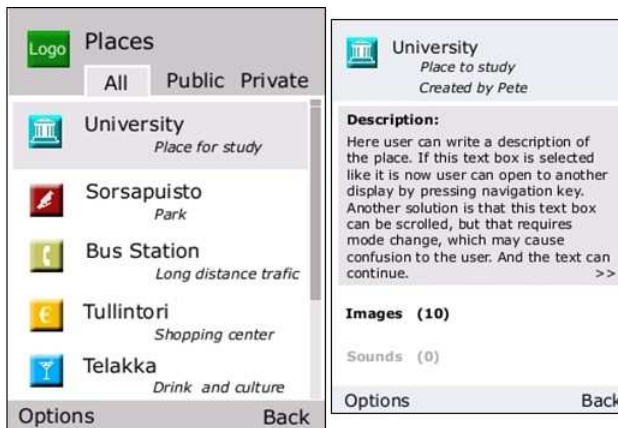


Figure 45: Screenshot of the system.

14.3 Methods and tools

The tools used in project are many, for management and for programming activities. For managerial tasks the group used:

- Microsoft Excel - this program was the initial entry-point for all the working hours reports;
- Zoho project management tool (www.zoho.com) - a tool that organizes information and integrates into the real work of a project is important when managing a project. Zoho tools are easy to use and could have made a difference if had been seriously used;
- OurTown website - our website saved us all the trouble to administrate the Excel XLS files and saved a lot of effort when reporting hours. It also automated summarization of hours and allowed us to quickly visualize history of working hours. It also generates graphics which enhance the visualization of the data;

For the programming activities the tools used were:

- Eclipse 3.2 - the Eclipse IDE automated some of the programming and organized the code to some basic extent. ;
- Mupe framework (www.mupe.net) - Mupe framework is the core supporting technology for OurTown. This platform provides all API which OurTown is using to implement the required functionality. Although some problems were encountered when using Mupe, is it certain that this technology will realize great achievements and allow people to write programs faster for mobile devices;
- Mupe tools for Eclipse - automa[h]tion of some development procedures made the developers work a lot easier by skipping some boring (but time-consuming) steps in the test activities;
- TortoiseSVN version control system (<http://tortoisesvn.tigris.org/>) - concurrent version control was very important to keep track of changes in the source code and also to maintain back-ups of the project;
- Java 2 Micro Edition (Wireless Toolkit, etc.) - Mupe relies on J2ME, and it was useful to test and simulate real execution of the program with Sun's emulators and simulators. Use of Java allows portability to a wide range of host operating systems.

The Zoho website fell in disuse with time due to the creation of OurTown website. Zoho now offers a new feature to manage shared Excel XLS files which makes possible for any project to have a very simple, easy, and cheap way to report working hours.

14.4 Project phases

Under the waterfall model the project was conducted until the requirements specification phase. It was finished in December of 2006. After initiating the incremental model, several minor prototypes were developed, however the cycle of this method remained within the design, implementation, and informal testing procedures.

The first months were spent strictly on specifying the requirements and trying to acquire the necessary phones from the customer. In early January and February the delays to deliver the phones required the team to spend time only in designing activities, when the main goal should have been delivering a functional prototype. This fact compromised the evolution of the development under the incremental model's principles. Even so, the group continued carrying out implementation and designing activities.

Each developer worked on his own, although the design was frequently discussed in meetings or through e-mail, the implementation and testing was done individually.

The table 73 gives an overview of the project execution.

Table 73: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
7	WF	Sep.06	Dec.06	56	462.5	1.18
5	INC	Jan.07	May.07	140	448	0.64

14.5 Conclusions

The project has failed to achieve its objective to produce a functional and complete application. Several factors contributed to this result, mostly the lack of adequate conditions to develop the project. The schedule of the project has been seriously affected as was mentioned before, and the management in the second semester failed to motivate and lead the group to conceive a better prototype.

It was a very good experience to work with an unknown technology and in a situation where the client is very absent, giving a lot of independence and freedom, but also too much indefiniton and uncertainty.

This is a case that can teach some lessons related with close contact with customer, and where certain benefits from agile methods could have been taken. The failure of the project, however, is directly related with the

Table 74: Group effort by activity.

Acti- vity	Plan- ning	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	43	56	269	95	113	23	24	0	222,5	845,5

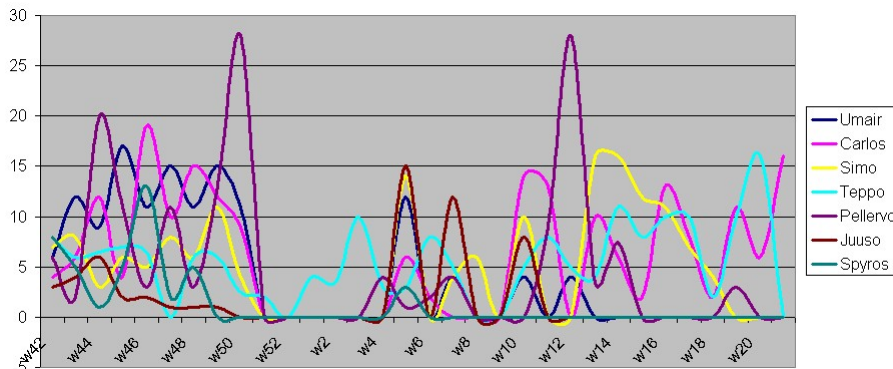


Figure 46: Working hours effort.

development model chosen. The team did their best and worked really hard as was possible to make the project a success.

Previous experiences with Mupe and Nokia projects already pointed out that the main problem in this project would be the commitment of the customer, but even so, the management was unable to prevent this risk from becoming a true problem.

14.6 Statistics

The project's working hours cannot be categorized due to the loss of some reports, however the totals for each worker during the weeks is intact. The following tables show the working hours and statistics for the working hours. In the sequence, the statistics of coding and source-code is provided by a set of graphics and tables.

The table 74 shows the working hours of the members divided into the categories. This can also be seen in a better in Figure 46.

As can be seen in table 75 the project produced only a few requirements and design deliverables.

Table 75: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
35	36	5	6	0	0

Table 76: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
14	0	0	2	3	2

That's because the design phase was quite delayed and there was little time to produce everything specified in the requirements document. Table 76 details the design results.

Table 77 shows the amount of documents and versions that were made. In total, there were 160 pages and in average 3 version of each document.

Table 78 below contains the statistic information about the source-code of the project. Most of the project's code is produced in XML-script, that is interpreted by Mupe client, so the code produce in Java language is really small. For this reason the codelines table for OurTown is modified.

The global project productivity metrics are shown in the table 8.

Table 77: Project's documents.

Document	Pages	versions
Project plan	18	3
Design plan	14	4
Requirements specification	21	4
Test plan	20	2
UI Specification	15	3
Test report	10	1
Project Story	7	2
Final Report	15	3
Weekly reports	31	
Monthly reports	8	
Review reports	1	
Total	160	62

Table 78: Project's codelines.

Source Files	45	Source Files
Directories	5	Directories
LOC	2977	Lines of Code
BLOC	248	Blank Lines of Code
Mupe Script LOC	1611	Lines of code in xml format for Mupe client
SLOC-P	479	Physical Executable Lines of Code
SLOC-L	322	Logical Executable Lines of Code
MVG	85	McCabe VG Complexity
C&SLOC	3	Code and Comment Lines of Code
CLOC	639	Comment Only Lines of Code
CWORD	2623	Commentary Words
HCLOC	56	Header Comment Lines of Code
HCWORD	399	Header Commentary Words

Table 79: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
5.99	496.98	1.75	5.84	11.02	26.71

15 Fot the Wok

15.1 Overview

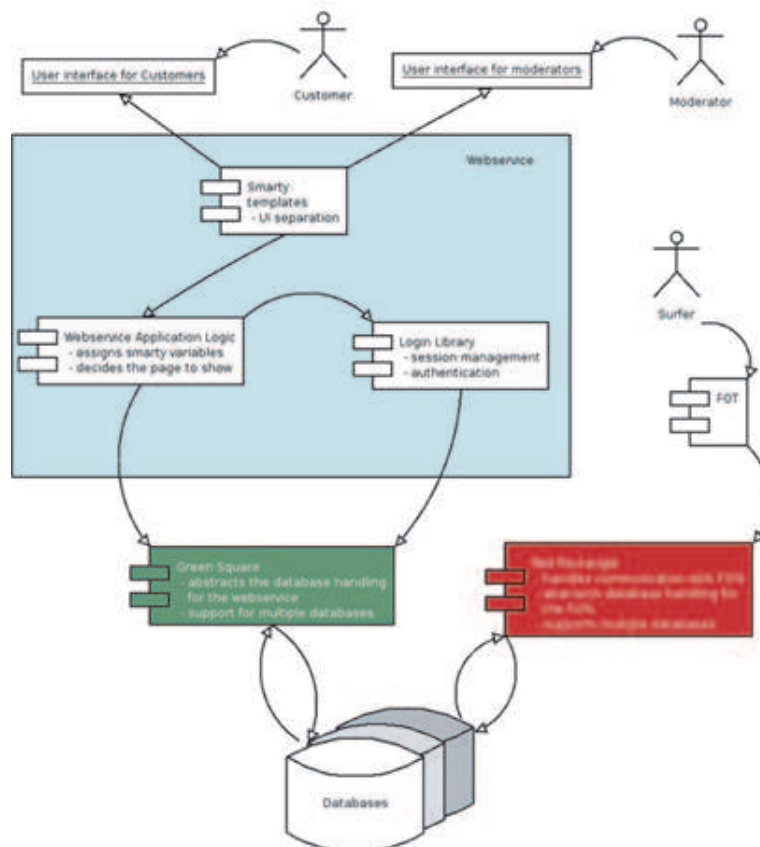


Figure 47: Overview of the project.

The goal of the project is to create a centralized feedback-system for websites. The system enables website owners to add a query form, also known as a FOT, to their website. The FOT will be presented in an unobtrusive manner and answering to the questions inside is totally optional. Once a casual web surfer answers the questions, the data is sent to a main server which stores the answers to the questions. Using the webservice available on the main server, the customer can then add questions to a FOT and view the answers to their questions. A customer is free to add questions to any website using the FOT-form, i.e. the customer doesn't need to own the website or have any kind of relation with it, the only prerequisite for adding questions to a specific website is therefore that the website in question has a FOT added

to it. Adding the FOT to the site must happen beforehand and naturally it will be done by the administrator or owner of the website. The limited amount of questions that appear on the FOT-form is determined internally by the system.

15.2 Organisation and management



Figure 48: group picture :)

Project managers coordinate and assign the tasks to the team. Team members will be divided into 2 teams, client side and server side. Project members can be shifted between 2 teams depend on situation. The client side team will focus on FOT, the part that appear in web pages and how the to communicate with the server side. The server side will focus on database and web services for web administrator. Usability expert will cover all issue that related to user interface.

15.3 Methods and tools

During the first meetings we quickly decided to use an iterative development model. It worked rather well in the beginning with longer iterations and clear targets. During 2006 we held weekly meetings. Sometimes twice. The average meeting would last around one and a half hours, during which

we'd appoint tasks for the next week. After New Year once we began the real development effort we pretty much stopped doing our default two week iterations and moved into one week iterations that were phased by our development meetings. The development meetings also acted as our weekly meetings. In the beginning of the year we still did our iteration plans, but decided to quit those since no valuable feedback was received from the clients and the only thing those iteration plans helped with was the structuring of tasks which were quite clear all along. Instead of using our efforts on those plans we focused on updating the design plans and test plans. Each week we held a development meeting on Thursday beginning from three and lasting until seven. The attendance of the meetings varied, but usually most of the members were there. The development meetings were a huge help to development, boosted morale and balanced the working amounts. Our testing procedures were quite simple: we took the test cases devised from our use cases and went through them. Some testing was already done during March. Here a Tool list, which used in this project.

A variety of editors were used. Including, but not limited to Eclipse, Kate, SKEdit, JEdit, Nano and Vi. Those who did not use Eclipse or an editor that did not have SVN-support needed to use something for Subversion access. Those working on Windows generally used TortoiseSVN, which integrates quite nicely to the Windows Explorer menu. Some preferred the use of the command line client to SVN. Using subversion as the central repository was a good choice. A lot of the time we had to rely on SVN's merge capabilities since the developers were working on the same files. However these situations were nicely solved thanks to SVN. Thanks to the vast amount of different tools available, using SVN was no problem in our multi-platform environment.

15.4 Project phases

There are four phases in this project. Inception phase started and end in October . Only important document in this preliminary analysis which is also the milestone for ending the inception phase. Elaboration phase is during November and December . In this phase, we already started some coding and also design at the same time. The iterations started in this phase as well. There are two iterations in this phase. Construction phase started in January and last until March. Note that this is where we begin to off the schedule from preliminary analysis. According to plan, we should finish construction phase before March. However, there were some recode and technical difficulties which make construction phase last longer than it should be. All functionalities were implemented at the end of March, and that was



Figure 49: Main screen of the Fot the Wok system.

milestone for ending construction phase. Transition phase is in April. In this phase. We focus in bug-fixing, and implement the user interface. So in conclusion, this project’s schedule is different from first schedule version for one month. The delay occurred in construction phase.

15.5 Conclusions

Evolution method is excellent software development method for project that do not solid requirement from start of project. We have new minor design plan almost every time we met client until everything is in place. This model will let us start coding to the very beginning of the project. Everyone in team and also client can see the early prototype and suggest us to fix or if something is wrong. Development day is also crucial for our project group. In construction phase, most work is done in development day (more that 80% of total working hours).

15.6 What to do better next time

Major problem in this project is team members working hours report. Many members always forget to fill in their working hours, this make total working hours inaccurate. We still have to find better solution for this problem.

15.7 Statistics

Table 80: General project information

Team	Dev. mod.	Start	End	Days	Hours	Hours / (Days * Team)
2+5+1	EVO	9.10.2006	24.4.2007	199	759	0.48

Table 81: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	126.5	82.5	40	70	284	-	52	22	82	759
Usab.										78
Total										837

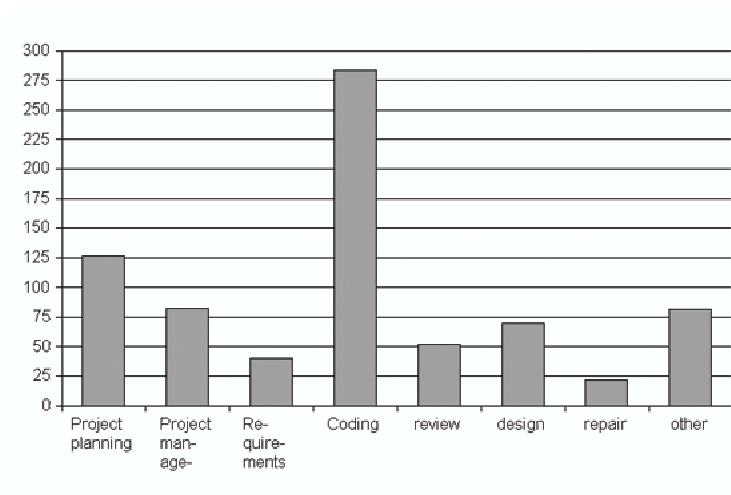


Figure 50: workinghours by tyoe-month :)

Table 82: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
23	33	-	12	1	4

Table 83: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
24	1	3	1	-	2

Table 84: Project's documents.

Document	Pages	versions
Preliminary analysis	16	8
Project plan	31	6
Requirements specification	23	9
Design plan	24	5
Test plan	16	3
User's guide	3	1
Final report	16	4
Final story	8	3
Weekly reports	24	
Monthly reports	7	
Total	168	

Table 85: Project's codelines.

Language	JAVA
LOC	7821
SLOC	3227
Reused code	-
Reused and modified code	-

Table 86: Productivity metrics related on project months (PM).

PM	LOC / PM	Files / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
4.95	529.351	6.02 5	7.024	9.432	51.50

16 Extension Of SysMLL

16.1 Overview

The main goal and purpose of *Extension of SysMLL* (System for Multiple Language Learning) project was exactly what the name suggests and that is to add more functionality in this web based multiple language learning system that is in use in the department of School of Modern Languages and Translation Studies (Russian) of University Of Tampere since 2006. Initial or the base of the system was developed by year 2005-2006 project group and by the client. Our task was to, add multi media based functionality to current system, such as image to meaning data and sound to words data. Other functionality we added and improved are: separation of word finding and saving it to different meaning, some usability problem in the old system, adding system parameter so that some process value can be accessed dynamically. Although our client had many useful requirements in his list, we had shortened his list due to time and resources.

16.2 Organisation and management

The client of this project is department of School of Modern Languages and Translation Studies (Russian) of University Of Tampere and representative was Dr. Mihail Mihailov, lecturer in Russian language and Translation Studies. There were two project managers and three developers. They are: Kamrul Ahsan (Project manager), Majid Ali Khan (Project manager), Pipsa Korte (Developer), Maria Salo (Developer), Lily Lin-Marsalo (Usability Expert). Two project work course students quit the course.

16.3 Methods and tools

Team used followings major tools in different development phases of the project:

- REM (Requirement Management) Version 1.2.2 (Pre-release)
- MS Visio
- PHP (Version 5.2.0)
- PostGreSQL (Version 8.1.5)
- WinSCP

- PuTTY / Secure CRT (Version: 0.53)
- SSH Tectia Client (Version: 4.2.0 (Build 22))
- Joes Own Editor (Version: 3.3)
- Notepad++ (Version 4.0.2)

Team also used its own design document, DFD (Data Flow Diagram) diagrams and pseudocode in different design phases of the project. General purpose PHP coding conventions were used when ever possible during coding phases. Due to small team size, we have not used any version management system or software for coding.

16.4 Project phases

We used incremental project management development methods. There were three increments in the whole project. Choosing such methods was quite effective decision. First of all project was extension of already developed project. So we just needed to discuss new requirements and implement those via planned increment. Second of all, our client was very active and he had very good understanding of the requirements and system in general. Third of all, both of our team members had good knowledge about programming environment and client/server coding.

We had outsourced our first increments testing part to USVA project team. There was also one big full-system usability test that was lead by our usability expert Lily Lin-Marsalo. In that test there were four volunteer participants who evaluated the system more than thirty minutes each. Team also co-operated with *On line help* team was organized by Technical Communication Programme of the School of Modern Languages and translation studies and the name of the course was *Online Instructions*, They have created total help system that is now integrated into SysMLL. So there were two major co-operative works.

Followings are some GUI (Graphical User Interface) print screen that was taken from the current system. Note that these functionalities were developed by this team.

16.5 Conclusions

Although we lost two team members at the beginning of our project, determination and hard work of rest members made this project a successful one. Team not only finished its home work (Implementing and integrating

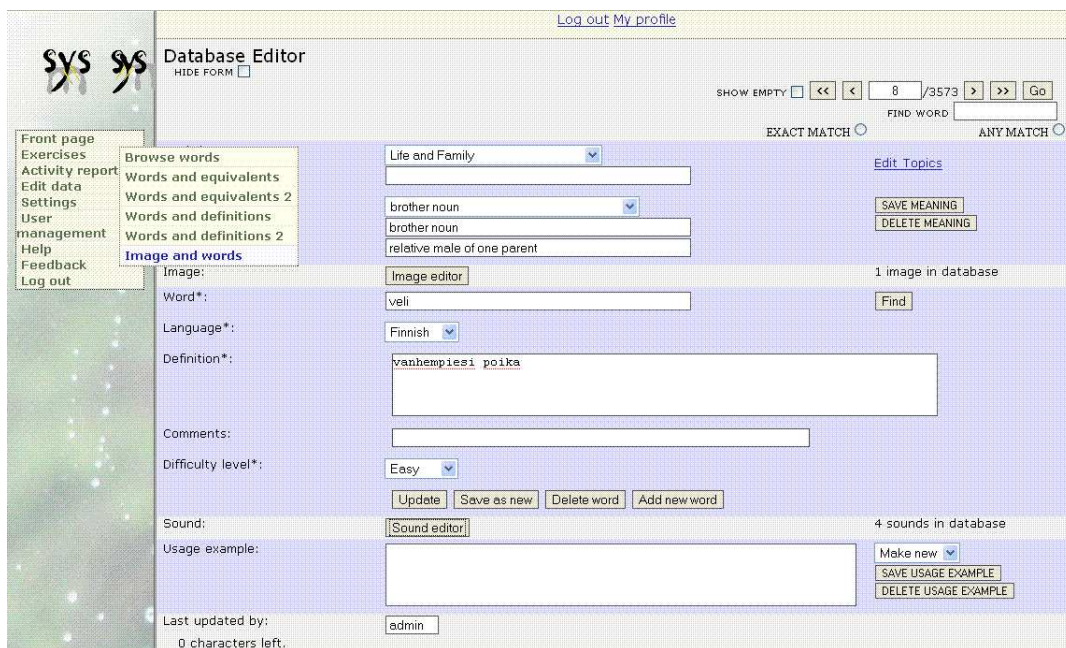


Figure 51: Accessing “Image and exercise” function from Exercise menu.

Table 87: General project information

Team size	Dev mod	Start date	End date	Days	Hours	Hours / (Days * Team)
2+2+1	Incremental	6.10.2006	16.5.2007	220	1080	1.02

three increments requirements) but also worked successfully with two different external groups for two different occurrences. Team’s usability expert also tested the integrity and usability factor of the system with a big-bang usability evaluation. Now system is much more user friendly. Both project management and project work group members learned many practical issue of developing web based software in real environment. We have also gained many co-operative work related knowledge and experience, including outsourcing project milestones. We learned how to involve Usability issue at the beginning of the project and how it improves the quality of the product. Full system Usability testing also added one more learning dimension of our learning process. Overall, this project improved our confidence level to manage and develop web based software project.

16.6 Statistics

Table 88: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	37	229	120	91	130.5	105	9.5	9.5	239.9	1080
%	3.4	21.2	11.11	8.4	12.08	9.7	0.8	0.8	22.20	
Usability				80						80
Total	37	229	120	171	130.5	105	9.5	9.5	239.9	1160

Table 89: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
20	31	3	7	2	21

Table 90: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence giagrams	State diagrams	Other diagrams
48	4				11

Table 91: Project's documents.

Document	Pages	versions
Preliminary analysis	10	1
Project plan	30	7
Project's usability plan	21	4
Requirements specification	20	6
Design plan	48	11
User interface document	10	1
Test plan	32	3
User's guide	*	
Test report	12	3
Final report	42	7
Final story	10	4
Weekly reports	15	
Monthly reports	11	
Review reports	20	
Total	281	

Table 92: Project's codelines.

Language	PHP/CSS/JavaScript
LOC	2616
SLOC	1686
Reused code	631
Reused and modified code	224
Files	3-5(PHP), 1(JavaScript), 1(CSS)

Table 93: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
7,6	342,9	0,9	8.9	6,9	43,7

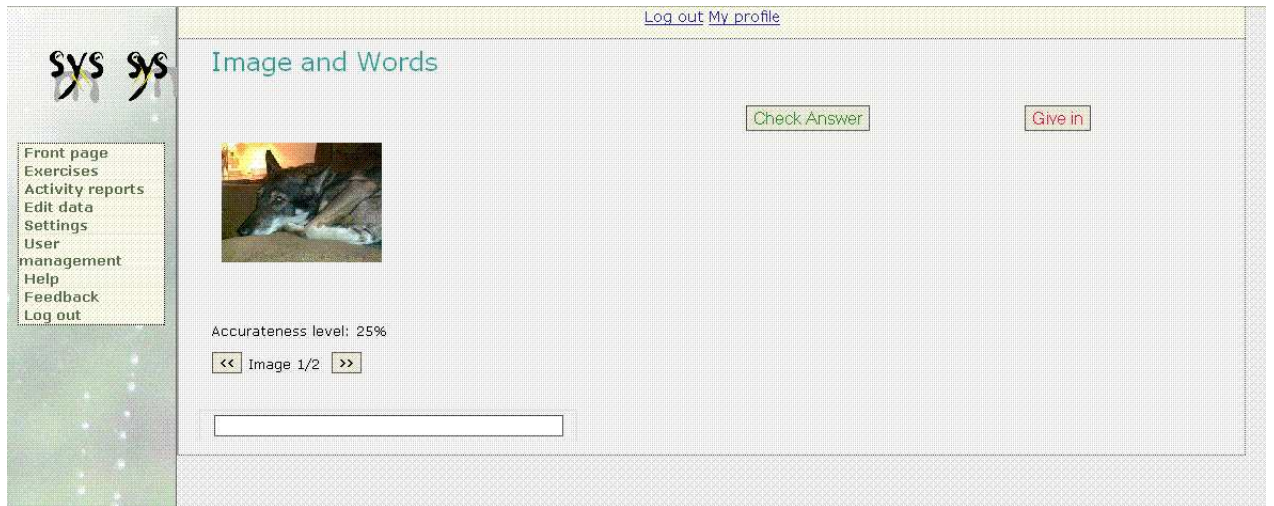


Figure 52: "Image and exercise" dialogue.

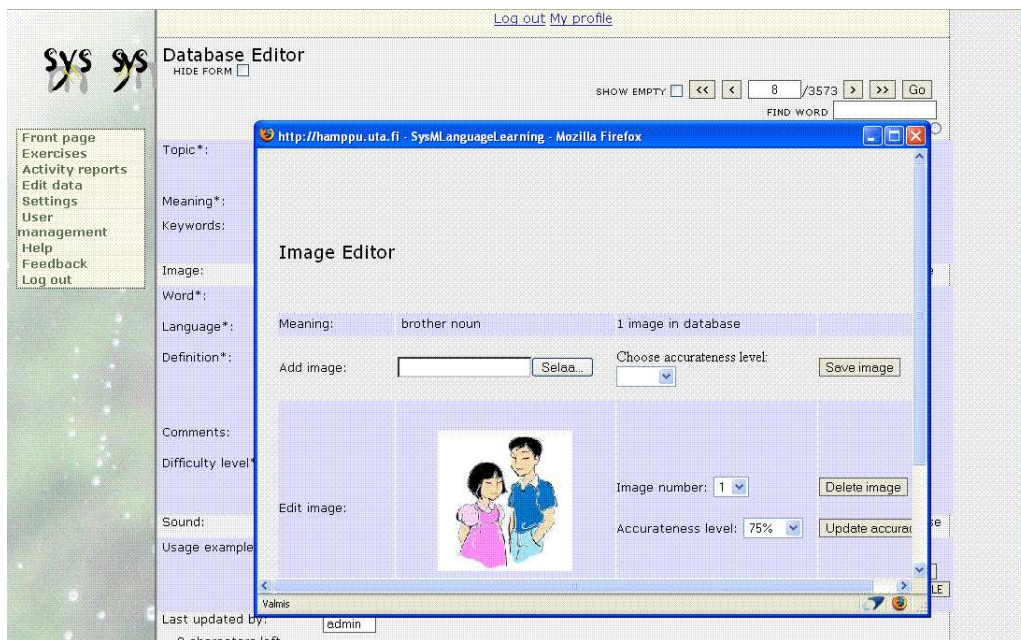


Figure 53: Image editor for add/edit/view image for subjected meaning data.

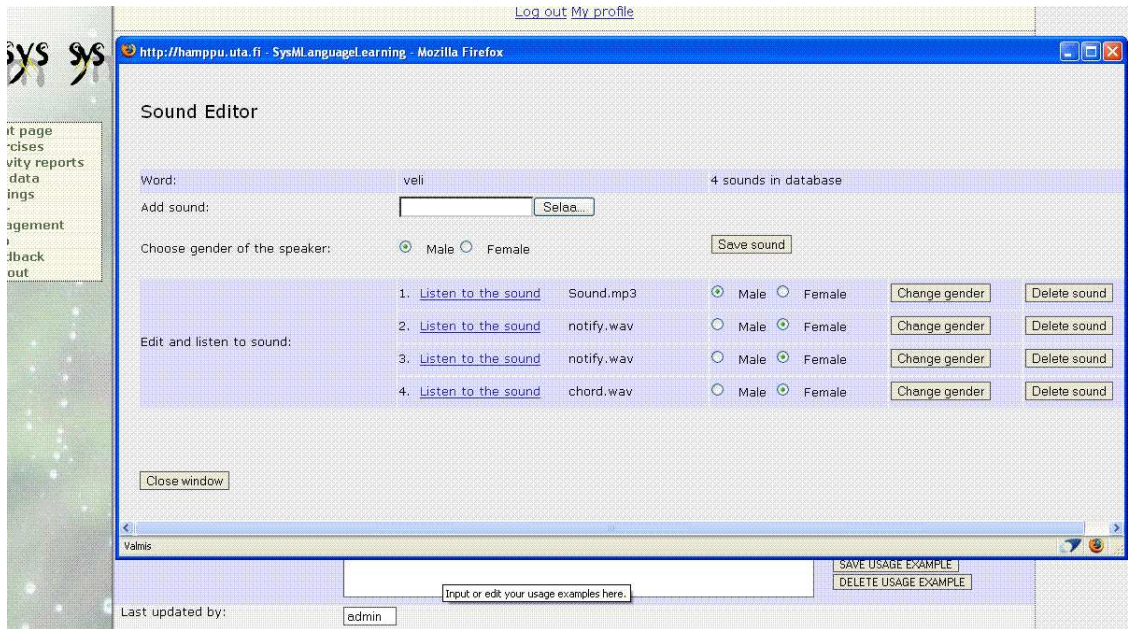


Figure 54: Sound editor for add/edit/play sound data for subjected word data.

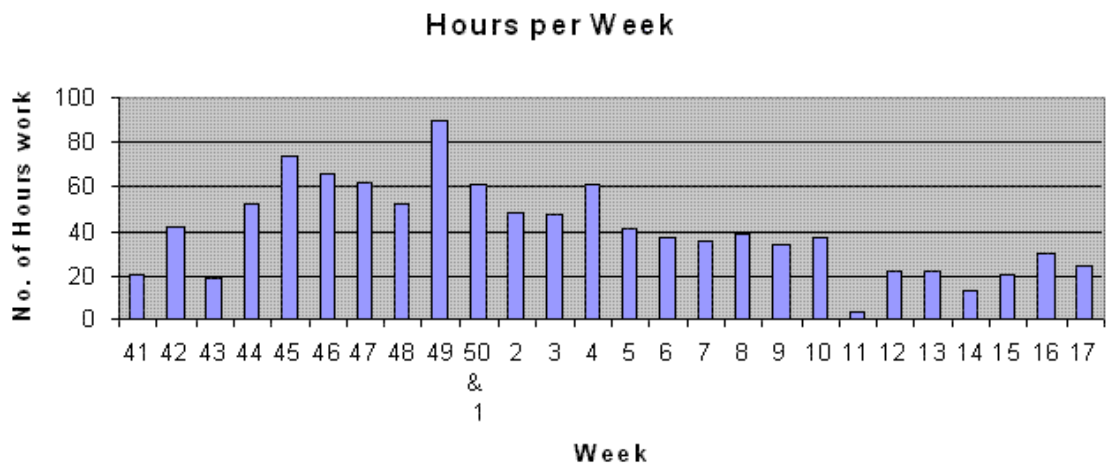


Figure 55: Weekly workload of the project.

17 USVA

17.1 Overview

USVA system is designed for supporting the analysis of usability test recordings that are recorded in digital format. The analysis consists of two parts and the USVA system supports them both; watching the test recordings and scoring the media file and analyzing the usability observations that are made to discover usability problems and make a report of the findings of the test.

The address of our project is: <http://koti.mbnet.fi/lim/usva/index.html>.

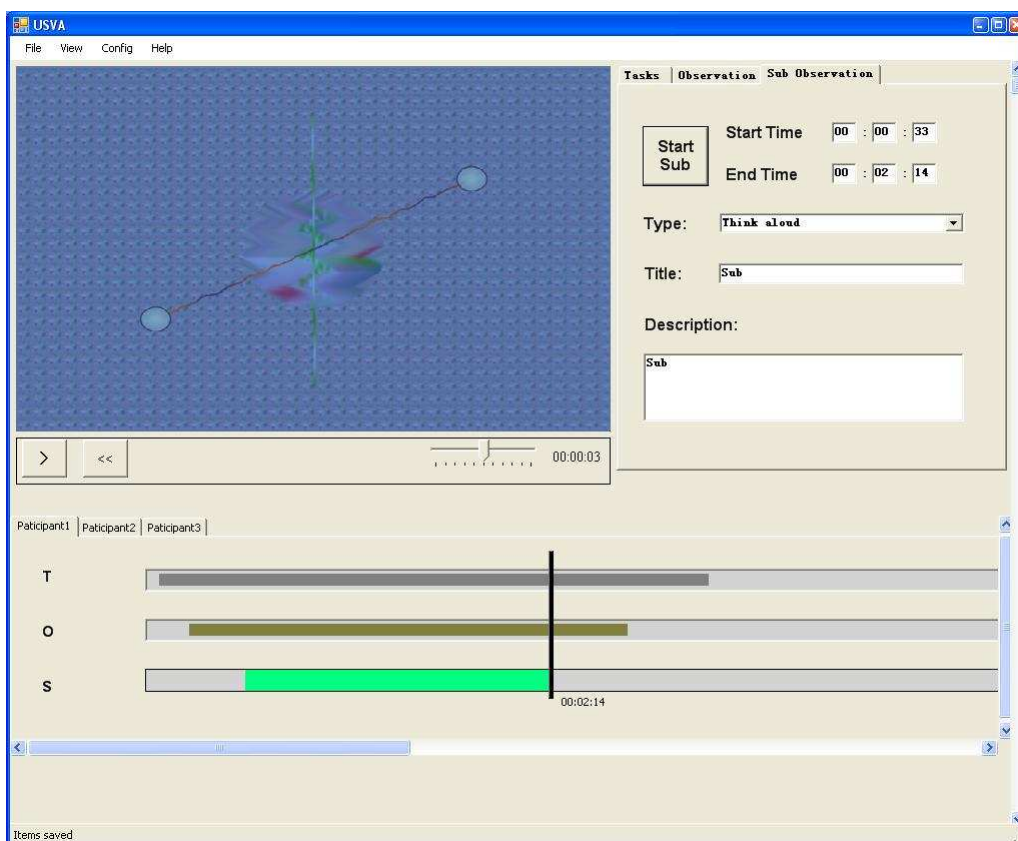


Figure 56: USVA screenshot.

17.2 Organisation and management

The project group consists of 8 students of the University of Tampere. There are 2 project managers who are managing the project as a part of their software project management course, 5 group members from the project work course and 1 member from the usability group of the same course.

Two project managers Ghassan Humadi and Linfeng Li divided their job, so that Linfeng was project chief manager in autumn 2006 and Ghassan is project chief manager in spring 2007. In autumn Ghassan was project vice manager and Risk manager and in spring Linfeng was project vice manager and Technique assistant.

- Requirement officer: Jarno Ojala
- User interface officer and web site maintainer: Lauri Immonen
- Testing officers: Jouni Laitinen and Outi Tuisku
- Testing officers: Jouni Laitinen and Outi Tuisku
- Technical officer: Peng Wang
- Usability officer (Usability team): Markku Myllylahti

17.3 Methods and tools

The tools used in our project are listed below.

- Microsoft Visual Studio 2005. Purpose: Used for software development, writing and compiling the source code.
- UMLet. Purpose: Used for visualizing software design.
- Concurrent Versions System (CVS). Purpose: Used for project documentation version management.
- Microsoft Excel. Purpose: To calculate and summarize the working hours of group members.
- GanttProject. Purpose: Used for project scheduling with Gantt Chart and resource load chart.
- Moodle forum. Purpose: Used to discuss the project management issues.



Figure 57: Project members.

17.4 Project phases

In our project, we followed the incremental model instead of our proposed ones. Therefore, the whole project development comprised four phases, primary analysis, requirement collection, increments development, and testing.

We spent about 2 weeks to conduct the primary analysis. In this way, we got to know the project scope. After that, we gathered the requirements from the requirements draft and the meetings with our clients. Then we began to develop the application. When one target increment was ready, we kept testing and debugging and moved to the next increment.

The detailed project phases can be found from Figure 58.

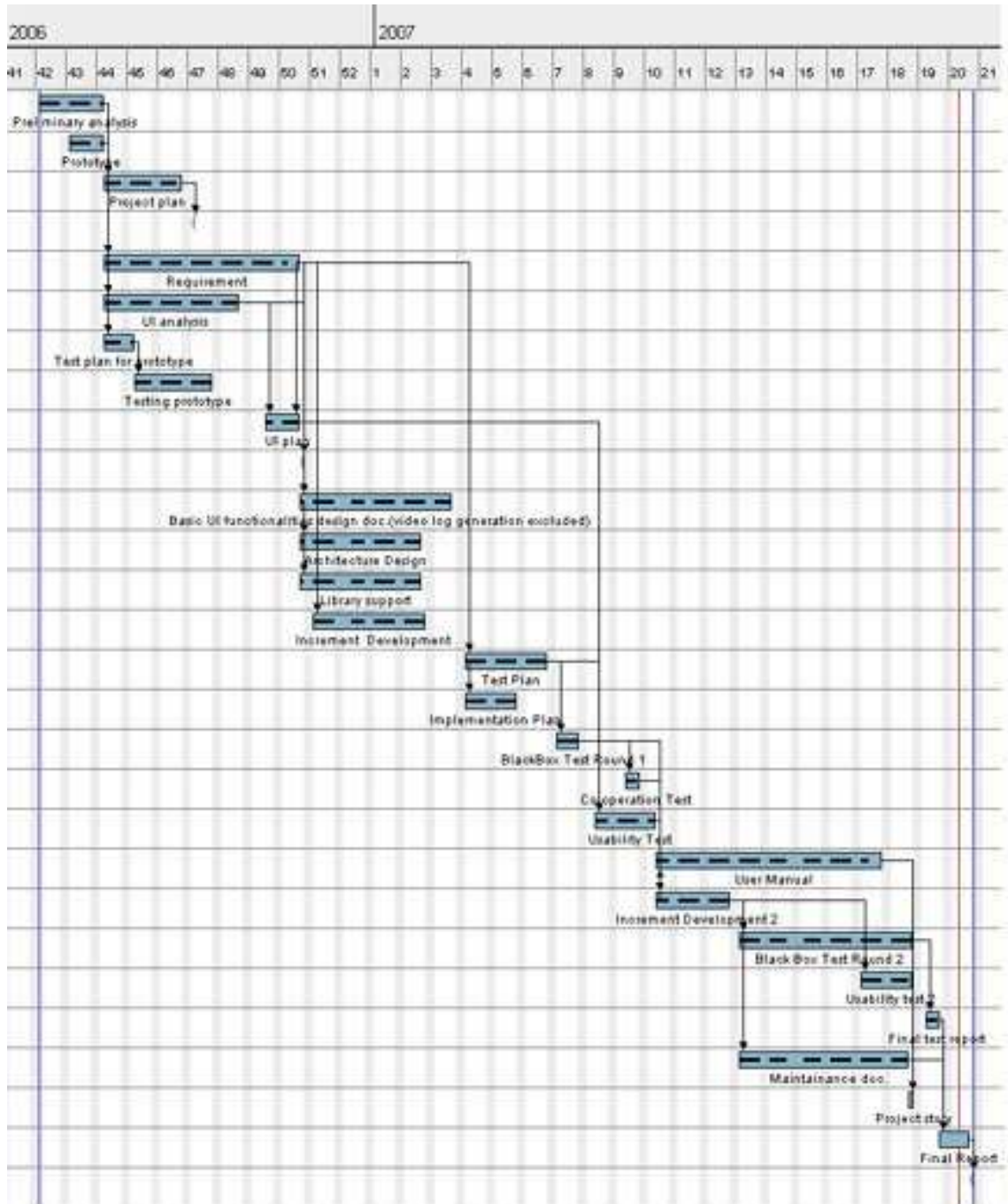


Figure 58: Project phases.

17.5 Conclusions

The project members collaborated with each other quite well. The developer could implement major requirements on time. The other members carefully designed and conducted various tests. Because of the efficient communication between each member, in general, we co-operated fluently and completed the tasks successfully.

17.6 Statistics

The total number of working hours is more than 1650 hours, including the usability team member. The weekly working hours chart is shown in Figure 60.

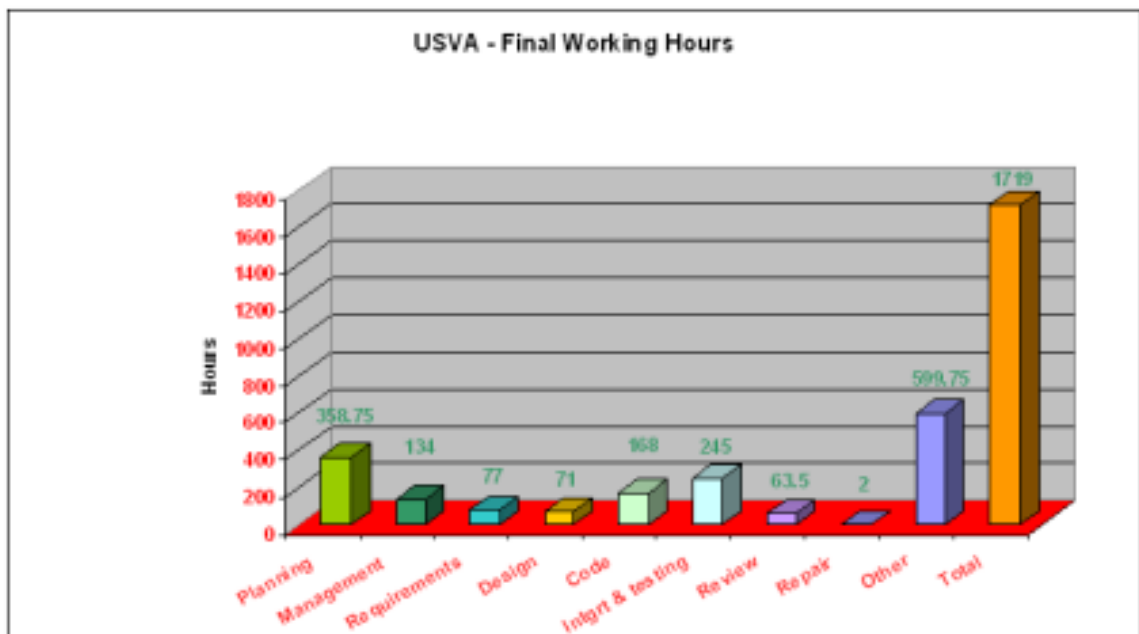


Figure 59: Weekly workload of the project.

17.7 Acknowledgement

Thanks a lot for our supervisor, Mr. Timo Poranen, our clients Tomi Heimonen and Jenni Anttonen. In addition, thanks for the help from the instructor of on-line co-operation, every co-operational group (Usability Team, SysMLL, SPOT, and on-line documentation groups) member, as well as all the members in our group.

Table 94: Working hours

Type	Hours total with Markku	Hours Total without markku
Project planning	358,75	282
Project management	134	129
Requirements specification	77	71
Design	71	69
Code	168	168
Integration and testing	245	185
Review	63,5	49
Repair	2	2
Other	599,75	536,75
Total	1719	1492,25

Table 95: General project information

Team size	Dev mod	Start date	End date	Days	Hours	Hours / (Days * Team)
2+5+1	Incremental	17,10,2006	21,5,2007	217	1719	0.99

Table 96: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	358,75	134	77	71	168	245	63,5	2	599,75	1492,25
%	20,87	7,8	4,48	4,13	9,77	14,25	3,69	0,12	34,89	
Usability	76,75	4,5	6	2	0	60	14,5	0	63	226,75
Total										1719

Table 97: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
51	58	15	12	-	-

Table 98: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence diagrams	State diagrams	Other diagrams
52	3	3	0	0	1

Table 99: Project's documents.

Document	Pages	versions
Preliminary analysis	9	1.1
Project plan	17	1.8
Project's usability plan	17	1.1
Requirements specification	20	1.8
Design plan	35	1.4
User interface document	17	1.1
Test plan	29	1.1
User's guide	27	1.1
Test report		
Final report	19	1,0
Final story	3	
Weekly reports	29	
Monthly reports	7	
Review reports	10+	
Total	239	

Table 100: Project's codelines.

Language	C sharp
LOC	4600+
SLOC	4200+
Reused code	200
Reused and modified code	1000
Classes	8
Functions	39

Table 101: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
11	418,18	0,72	5	8,36	31,9

18 IOI2006

18.1 Overview

Purpose of this project was to extend the functionality to the previous year's project IOI2005+. International Olympiad in Informatics (IOI) system's software is a web based solution so it can be used from anywhere where Internet is available. The software will run on the server, which is owned and maintained by Department of computer sciences of Tampere University.

International Olympiad in Informatics (IOI) is a competition in computer sciences for secondary school students. It is an annual event and it is organized every year by one of participating countries. As the name of the competition says, it is an international event and its participants are from all over the world. The case has been before that every year the hosting country has implemented its own administration system to serve the competition. This is not very wise, because unnecessary efforts have been used every year to build up a new system. This product is meant to become a system that will serve the competition every year in the future.

Our purpose was to re-use old usable components and objects and design and implement the missing ones. The product has already been developed for two years in a row, but still few important features are missing. Hence our project aimed at implementing these features to deal with automating travel and accommodation allocation. In the end only the first increment, a bus allocation feature, was implemented.

18.2 Organisation and management

At the beginning of the project, project group had 6 members including two project managers and we also had a member of the usabilityteam. Due to the resignation of the both project managers currently we have 4 members and no project managers and also our usability member has not been involving to the project. After these resignments Outi Rähkä took responsibility to be our contact person.



Alpertti



Riikka



Antti



Outi

Current members are:

- Alpertti Tirronen
- Antti Jämsen
- Outi Rähä
- Riikka Valtanen

Client of the project is Jyrki Nummenmaa. He is the head of the Department of Computer Sciences at the University of Tampere and has been the member of the IOI Scientific Committee. Timo Poranen was a Course Instructor and supervisor. We also received help from Pauli Savolainen, who had made many functionalities to the previous version.

18.3 Methods and tools

The project group used the following tools in developing the product:

- Eclipse IDE 3.1.1
- Java 1.5.0
- Web Tools Platform 1.0 (WTP) plug-in for Eclipse
- Fujaba 1.4.1 + DBSchema plug-in
- Tomcat 5.5
- Struts 1.2.9
- PostgreSQL 8.0

18.4 Project phases

YEAR 2006

October:

- Requirements elicitation from client
- Preliminary analysis preparation

November:

- Project plan development

- Software Requirement Specification (SRS) development

- Training of technologies to be used in project

December:

- Software Requirement Specification (SRS) development continues

- Training of technologies continues

YEAR 2007

1st January - 15th February:

- Design of the system and training of technologies

16th February - 31st March:

- Design of the system and training of technologies

- Integration with old system

April:

- Design of the system and implementation

- Integration with old system

- Integration testing

May:

- Implementation and integrating

- Testing

18.5 Conclusions

The project IOI2006 was a good experience to all of the group members. Project took a lot of everyone's time but it gave us all a lot of valuable knowledge and skills. We also learned that working in an international environment could cause surprising problems beyond imagination. After all, how many times one hears of both of the project managers resigning at school or even university environment, especially when there is an actual customer expecting some results from the project.

The project started well but soon we only wrote documents without planning or discussing about the matters and reasons why these documents were done. We should have started working together and have more specific meanings to the meetings. Also we should have had more meetings with the client and have much more communication between the project managers and the group. Thus, we learned a lot of organizing and project management skills during the course of the project.

18.6 Statistics

Table 102: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
0+4+0	WF	10.10.2006	16.5.2007	219	764.5	

Table 103: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	325.5	49.5	25	47.5	74.5	35	13	14	180.5	764.5
%	42.6	6.5	3.3	6.2	9.7	4.6	1.7	1.8	23.6	100

Table 104: Requirements and high-level design outcome.

Pages	Use-cases	UI screens	Database diagrams	Database tables
21	16	10	1	3

Table 105: Design outcome.

Pages	Overview diagrams	Class diagrams	Other diagrams
46	5	8	8

Table 106: Project's documents.

Document	Pages	versions
Preliminary analysis	8	1.0
Project plan	20	1.1
Project's usability plan	8	1.2
Requirements specification	21	1.2
Design plan	46	1.1
Test plan	6	1.1
Maintenance document	36	1.0
Test report	3	0.1
Final report	10	1.0
Final story		
Weekly reports	17	
Total		

Table 107: Project's codelines.

Language	JAVA
LOC	4382
SLOC	2627
Reused and modified code	746
Files	36
Modules	1

Table 108: Productivity metrics related on project months (PM).

PM	LOC / PM	Files / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
7	626	5.1	9.6	12.5	

19 CALESYNC07

19.1 Overview

The goal of the Calesync07 project is to develop a calendar application that uses the IP Multimedia Subsystem (IMS) architecture in order to communicate with peers and application server. The application itself is designed to allow users to propose and arrange appointments through the initial availability check followed by suggested and agreed meeting times. The Calesync07 project continues the work of the CALEX2006 project, which was ended in the spring of 2006.

19.2 Organisation and management

The CaleSync07 project group consisted of seven team members; one project manager, five developers, and one usability expert. Refer to table: 110.

The team's home page: <http://www.calesync07.com/>

19.3 Methods and tools

- Project weekly meeting.
- Email informs most of the project events and tasks.
- Web site, which served as the central archive for all documentation regarding the project, including working hour observation.
- SVN repository, was used for managing project source code. The URL can be <http://svn.calesync07.com>.
- Borland C++ BuilderX Mobile Edition, it is free of charge, and it had been proven somewhat functional and capable by the project groups of previous years.
- S60 SDK 8.0a, S60 Platform SDK for Symbian OS.
- SIP SDK 4.0, SIP plugin solution for developers.
- 6680 mobile terminals, Nokia 6680 (SIP Stack 4.0, Symbian 8.0a) The Nokia 6680 device incorporates the Symbian operating system (OS) 8.0a. 6680 was the primary development hardware for CaleSync07.

19.4 Project phases

- October 13th: The initial meeting of the project team
- November 2nd: The project kick-off meeting as a teleconference with the representatives of the client in both Tampere (Finland) and Budapest (Hungary), as well as the BUTE project team.
- December 1st: Project plan review
- December 13: Project plan review with the client during steering group meeting
- December 13th: The presentation of the project at the university
- December 26th: Requirements specification initial version
- January 10th: Requirements specification review
- January 21st: UI Specification
- February 1st: Architecture design plan
- March 16th: Design plan review
- April 2nd: Test plan
- May 16th: The final presentation of the project at the university
- May 31st: The delivery of the final product to the client
- May 31st: The feedback meeting. Close the project

19.5 Conclusions

During this project, we concluded - as did the predecessors of this very project, namely CALEX2006 and Cook2005 - that contrary to normal practice in commercial projects within the frames of this course, we received very minimal, nigh nonexistent support from our client or collaborators. This was partly because one of the purposes of this project was to prove the ease of use and adaptiveness of SIP technology, and to show that even students could reach an acceptable outcome in a project related to this technology.

We had the pleasure and privilege of being part in a project which involved not just people from our own department, but also a project team from Budapest University of Technology and Economics. The whole project

Table 109: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
1+5+1	Incremental	02.11.06	31.5.07	175	921.75	0.75

Table 110: Project team

Name	Role	Responsibilities
Zhigang Yang	Project Manager	Management
Aidan Taus	Developer	UI, documentation, testing
Juha Maenalusta	Developer	UI, documentation, testing
Mikko Kokkonen	Developer	Web site, Implementation
Timo Helander	Developer	Implementation (UI, SIP)
Tung Doan	Developer	Implementation (SIP)
Lily Lin-Marsalo	Usability	Usability issue, document

was divided in two parts; server side (BUTE) and terminal side (UTA). To our knowledge, this has been closest to a 'real life' project that our department has hosted thus far. Because of this, we would like to express our gratitude for the opportunity we were presented with. This project was very challenging, which had both good and bad sides to it. On one hand, a challenging project is an extremely good opportunity to learn, but on the other hand, if it turns out to be too challenging, as in our case, it might lead to despair and frustration in certain areas. By this we're trying to say that by balancing the level of challenge of the project with the level of knowledge within the project team(s), the experience as a whole would be more fruitful for both the team(s) and the client.

When it is acknowledged that the project in question requires a certain level of knowledge and use of newer or otherwise less known technologies, consulting/help should be sought for and acquired in an as early phase as possible. Otherwise the project as a whole might be jeopardized, since the lack of experience and know-how always results in the postponement of deadlines, which might cause stress and lack in motivation. Estimations about the workload and possibilities to complete the project with what resources are available should be realistic, maybe even slightly exaggerated, and be inspected in a careful and humble manner.

19.6 Statistics

Table 111: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	72	60	107	107	437	90	49	0	0	922
%	7.8	6.5	11.6	11.6	47.4	9.8	5.3	0	0	100
Usability										-
Total	72	60	107	107	437	90	49	0	0	922

Table 112: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
46	40	6	12	0	0

Table 113: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence giagrams	State diagrams	Other diagrams
23	2	10	0	0	0

Table 114: Project's documents.

Document	Pages	versions
Preliminary analysis	15	1.0
Project plan	48	1.2
Usability plan	23	1.0
Requirements specification	23	1.2
Design plan	23	1.2
User interface document	23	0.9
Expert evaluation report	30	1.0
Test plan	23	1.0
User's guide	10	1.0
Maintenance manual	15	1.0
Test report	10	0.1
Final report	19	1.0
Final story		
Weekly reports	29	
Monthly reports	7	
Total	283	

Table 115: Project's codelines.

Language	Symbian C++
LOC	7351
SLOC	6190
Reused code	0
Reused and modified code	1160
Classes	30

Table 116: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
6	1292	5.3	12.3	26	72

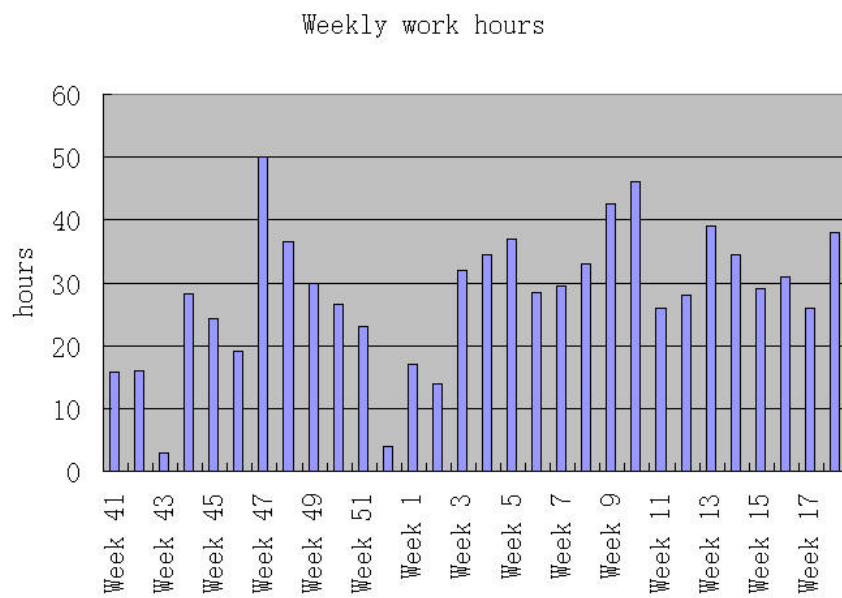


Figure 60: Weekly workload of the project.

20 DICE Corpus

20.1 Overview

The purpose of the project is to develop a web-based corpus which gives users an easy possibility to use free available texts for linguistic research. It includes electronic versions of different types of English texts from the 16th century to the present day.

The DICE Corpus is a web page with its own concordance program. It enables users to perform searches online from a predetermined set of Internet addresses of historical texts. With its own search engine, the users are able to perform searches of words or phrases from the texts, with the possibility of concentrating e.g. on specific periods, regional varieties or genres (Figure 63). By integrating the materials from different sources, scholars will not be faced with the task of searching these texts from the Internet and downloading them on the hard drives of their computers.

The search results will be presented on the screen as a list of concordances - one line for each hit. The corresponding URL addresses and brief details of the text resources will be given to the users in order to easily view the whole text if they wish (Figure 64).

20.2 Organisation and management

The DICE Corpus project group is comprised of 6 members, i.e. 2 project managers and 4 group members. The main roles divided into our group are managers, developers, and tester. The project managers are responsible for managing the project. The developers are responsible for designing and implementing the DICE Corpus system while the tester is responsible for testing the system. There is also one member from the usability team group involving with the project. The usability team member's responsibilities are to write the usability plan, user interface plan, and to do the usability test. The members of the project group and their roles are listed below.

- Project Managers: Thanyaporn Lerlerdthaiyanupap, Björn Bielesch
- Developers: Aleksi Vuorenmaa, Salla Kuisma, and Tom Eklund
- Tester: Mika Salmi
- Usability team member: Riki Kawakami

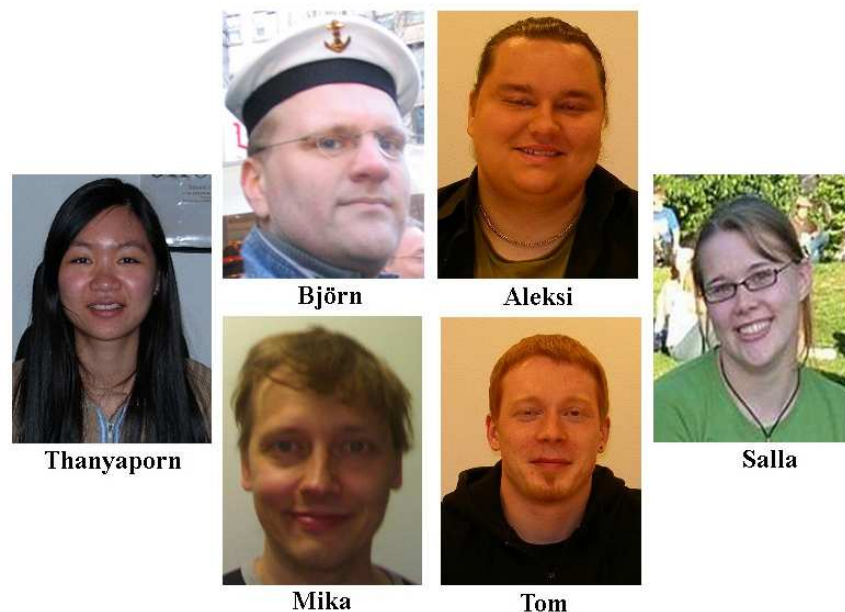


Figure 61: DICE project members.

The client of the project is Mark Kaunisto. He is a researcher at the English Department, School of Modern Languages and Translation, University of Tampere.

Moreover, we got the online help co-operation from students (online help team 3) who took the course Technical Communications Programme. The online help team's task is to produce online helps for the DICE Corpus system.

About project management, we organised the group meetings every week since the project started. The progress of the group was monitored through weekly meetings in which we discussed what we achieved in the previous weeks. We also followed everyone's weekly working hours and had a forum and a pbwiki site where we could distribute documents and information pertinent to the project. The project group communicated by using a mailing list which address is `dice-corpus@googlegroups.com`. The project's homepage is located at `http://dice-corpus.pbwiki.com`. All project documents are published there too.

20.3 Methods and tools

We used the following tools:

- Eclipse 3.2

- PHP MyAdmin 2.6.2
- HTML-Kit 1.0
- jEdit 4.2
- Google docs: used for updating group working hours. Sometimes, we used for sharing some parts of project documents.
- Google groups: used for internally communicating, e.g. sharing files, writing weekly memos, contacting group via e-mail, and posting messages for discussion.

20.4 Project phases

The development model of the project is the Iterative Development Model since the client was not sure about all requirements beforehand. The model offers a good way to react flexible on new requirements and changes within the development of the project (Figure 62).

According to our development model, there were four different iterations within our project. Each iteration deadline was on the 14th of Dec 2006, Feb 2007, March 2007, and April 2007. The last two weeks of April were used for bug fixing and finishing all necessary documents for the project.

The UI was developed according to the four iterations.

The online help was mainly written in April since it is based on the final version of the corpus.



Figure 62: Development model.

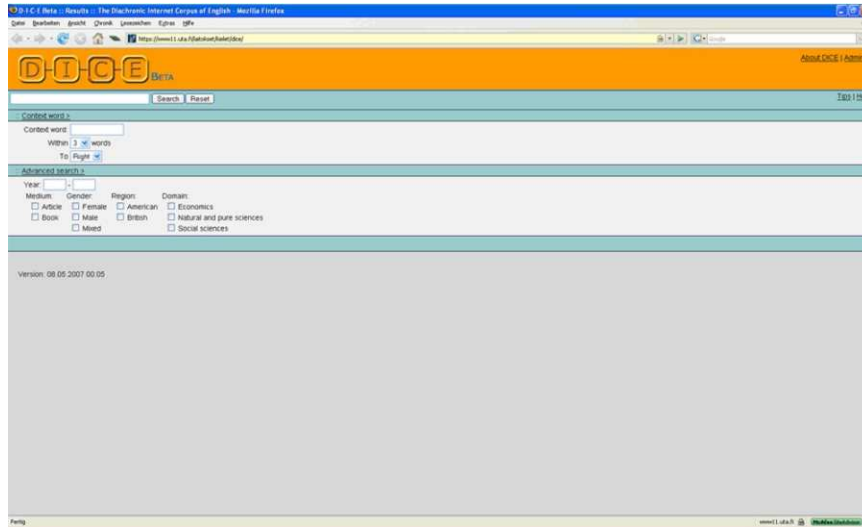


Figure 63: Main screen of the DICE Corpus.

Table 117: General project information

Team size	Dev. model	Start date	End date	Days	Hours	Hours / (Days * Team)
2+4+1	Iterative	10.10.2006	16.5.2007	196	1370.5	1.17

20.5 Conclusions

The project was really interesting and challenging so it gave us such a good experience. During the project, we encountered many problems such as no clear requirements from our client, passive members, some conflicts between managers but we figured out all those problems in the end. Overall, the project was successful from client's and our group's point of view because our client was quite happy with the end product. We all learned how to work in a project with a real client and real end users, and how to solve problems when we face them. Also, the project offered real development challenges and opportunities to improve everyone's own expertise, for example, programming skills, co-operation skills and project management skills. Finally, this project taught the importance of documentation and thorough project planning as the key role to the success of any project.

20.6 Statistics

The working hours per week are shown in the graph.

Table 118: Group effort by activity.

Activ.	Plan	Man- nage- ment	Req. spec.	Des.	Impl.	Int. and test.	Rev.	Rep.	Other	Total
Hours	297.5	150.5	52	74.5	382.5	97	37	26	254	1370.5
%	21.71	10.98	3.79	5.44	27.87	7.08	2.7	1.9	18.53	100
Usability										50
Total										1420.5

Table 119: Requirements and high-level design outcome.

Pages	Requirements	Use-cases	UI screens	Database diagrams	Database tables
53	29	0	20	1	11

Table 120: Design outcome.

Pages	Overview diagrams	Class diagrams	Sequence giagrams	State diagrams	Other diagrams
63	1	0	0	1	0

Table 121: Project's documents.

Document	Pages	versions
Preliminary analysis	10	1.0
Project plan	35	1.0
Project's usability plan	7	1.1
Requirements specification	29	1.0
Design plan	37	0.2
User interface document	24	3.0
Test plan	28	0.4
Online help	11	
Test report	10	0.4
Final report	25	0.1
Final story	6	0.1
Weekly reports	20	
Monthly reports	10	
Total	252	

Table 122: Project's codelines.

Language	PHP	JavaScript	CSS
LOC	3365	302	642
SLOC	2226	289	638
Reused code	0	0	0
Reused and modified code	24	0	0
Classes	0	0	0
Functions	34	10	0

Table 123: Productivity metrics related on project months (PM).

PM	LOC / PM	Classes / PM	RS & DES pages / PM	LOC pages / PM	Total pages / PM
7,5	574.53	0	12	11.49	33.6

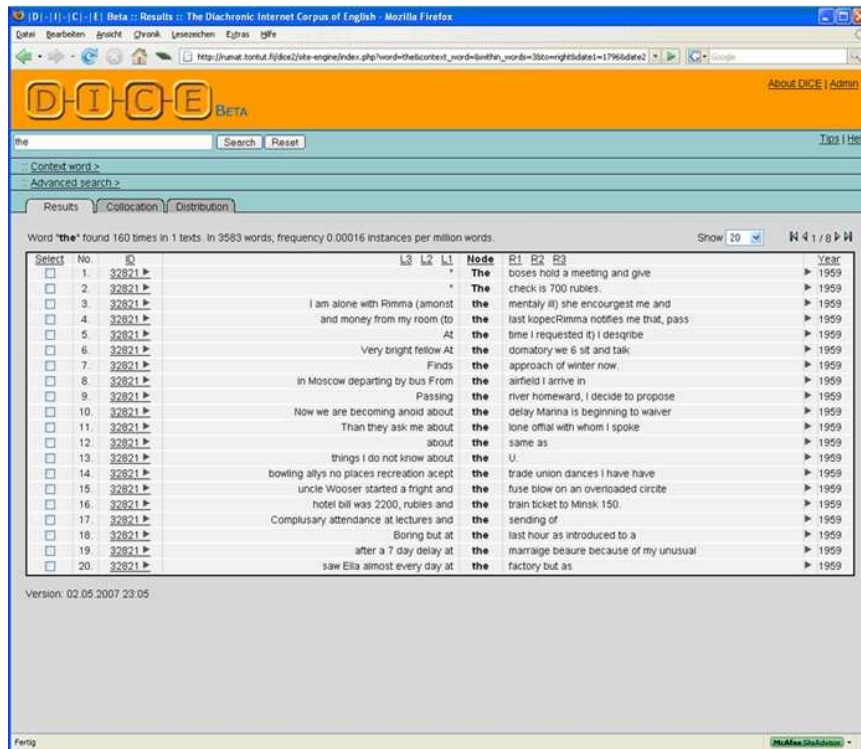


Figure 64: Screenshot search results.

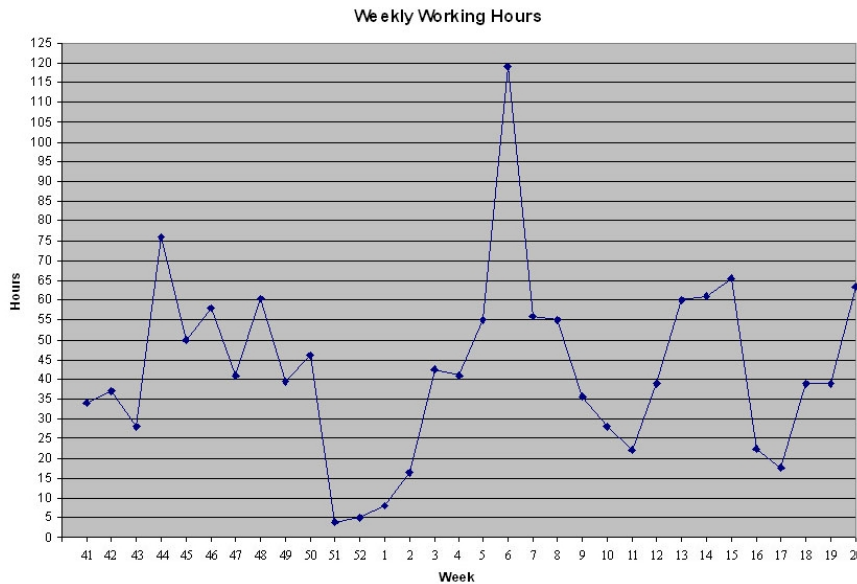


Figure 65: Weekly workload of the project.