

**Timo Poranen (toim.)**

# **Projektityöt 2006**



TIETOJENKÄSITTELYTIETEIDEN LAITOS  
TAMPEREEN YLIOPISTO

D-2006-6

TAMPERE 2006

TAMPEREEN YLIOPISTO  
TIETOJENKÄSITTELYTIETEIDEN LAITOS  
JULKAISUSARJA D – VERKKOJULKAISUT  
D-2006-6, KESÄKUU 2006

**Timo Poranen (toim.)**

**Projektityöt 2006**

TIETOJENKÄSITTELYTIETEIDEN LAITOS  
33014 TAMPEREEN YLIOPISTO

ISBN 951-44-6677-2  
ISSN 1795-4274

## Alkusanat

Tämä julkaisu sisältää Tampereen yliopiston tietojenkäsittelytieteiden laitoksen lukuvuoden 2005 - 2006 projektityö- ja ohjelmistoprojektin johtaminen -kurssien projektien loppukertomukset. Projektityöhön osallistui hieman yli 100 opiskelijaa, joista noin 20 tuli muista maista kuin Suomesta. Ohjelmistoprojektin johtaminen -kurssin suoritti 7 opiskelijaa, heistä useimmat toimivat projektipäällikköinä kahdelle eri projektille. Perinteisiä projekteja oli yhteensä 19 ja yksi projektiryhmä toimi "käytettävyyssryhmänä" tehden erilaisia käytettävyyteen liittyviä töitä muille projekteille. Keskimääräinen ryhmäkoko oli 5 projektityö -kurssilaista ja yksi projektipäällikkö. Projekteista 7 käytti työkielenään englantia.

Projektiryhmät kirjoittivat tähän julkaisuun lyhyen kertomuksen projektistaan. Kertomus sisältää kuvauksen aiheesta, läpiviennin pääpiirteittäin, käytetyt työkalut ja tekniikat, sekä ryhmän saamia opettavaisia kokemuksia. Jokainen projekti on koonnut myös tilastoja, jotka sisältävät projektiin käytetyn ajan jaoteltuna projektin eri vaiheisiin ja työn laatuun, sekä tuotetut dokumentit ja koodirivien lukumäärän.

Laitoksemme 19 projektista onnistui hyvin tai erittäin hyvin 10, 7 projektia menestyi tyydyttävästi ja kaksi projektia onnistui välttävästi, kun tarkastellaan prosessia, lopputuloksen laatua ja asiakastyytyväisyyttä. Hyvin menneille projekteille oli ominaista projektiryhmän hyvä ryhmähenki, osaava projektipäällikkö ja aktiivinen asiakas. Huonosti menneisiin projekteihin oli useimmiten syynä projektin jäsenen poistuminen kurssilta, mikä johti muiden projektin jäsenten työmäärän kasvuun ja joskus myös motivaation heikkenemiseen, sekä tekniset ongelmat toteutusvaiheessa. Käytettävyyssryhmän kokeilu oli kohtuullisen onnistunut, ja tulevina vuosina käytettävyyden huomioimista projektiopetuksessa tullaan lisäämään.

Lopuksi haluan kiittää Tampereen teknillisen yliopiston lehtoria Tero Ahteeta monista projektiopetukseen liittyvistä keskusteluhetkistä ja hyvistä ohjeista, sekä FT Isto Ahoa, joka loi vuosina 2002 - 2004 hyvän pohjan laitoksemme projektiopetukselle.

Tampere, kesäkuu 2006  
Timo Poranen

# Sisältö

<b>1 Usability team</b>	<b>1</b>
1.1 Yleiskuvaus projektista . . . . .	1
1.2 Projektioorganisaatio ja hallinta . . . . .	1
1.3 Käytetyt työkalut . . . . .	2
1.4 Projektin vaiheet . . . . .	2
1.5 Yhteenveto . . . . .	3
<b>2 IOI2005++</b>	<b>4</b>
2.1 Overview . . . . .	4
2.2 Organisation and management . . . . .	4
2.3 Methods and tools . . . . .	5
2.4 Conclusions . . . . .	6
2.5 Statistics . . . . .	7
<b>3 XML-Search Engine</b>	<b>8</b>
3.1 Overview . . . . .	8
3.2 Organisation and management . . . . .	8
3.3 Methods and tools . . . . .	8
3.4 Project phases . . . . .	8
3.5 Conclusions . . . . .	8
3.6 Statistics . . . . .	11
<b>4 Lindelöfin perilliset</b>	<b>13</b>
4.1 Yleistä . . . . .	13
4.2 Projektiryhmä ja työjärjestelyt . . . . .	13
4.3 Työskentelymetodit ja työkalut . . . . .	13
4.4 Projektin vaiheet . . . . .	15
4.5 Yhteenveto . . . . .	16
4.6 Tilastoja . . . . .	16
<b>5 SysMLL</b>	<b>18</b>
5.1 Overview . . . . .	18
5.2 Organisation and management . . . . .	18
5.3 Methods and tools . . . . .	18
5.4 Project phases . . . . .	19
5.5 Conclusions . . . . .	19
5.6 Statistics . . . . .	20

<b>6</b>	<b>Heuristisen arvioinnin tukijärjestelmä</b>	<b>22</b>
6.1	Yleiskuvaus ohjelmasta . . . . .	22
6.2	Projektiorganisaatio . . . . .	23
6.3	Projektin eteneminen . . . . .	23
6.4	Projektin hallinta . . . . .	24
6.5	Käytetyt välineet ja menetelmät . . . . .	25
6.6	Ongelmat . . . . .	25
6.7	Ruutukaappauksia järjestelmän näkymistä . . . . .	26
6.8	Tilastot . . . . .	26
6.9	Koodin määrä . . . . .	27
<b>7</b>	<b>DivXML Editor</b>	<b>28</b>
7.1	Overview . . . . .	28
7.2	Organisation . . . . .	28
7.3	Problems and risks . . . . .	29
7.4	Management . . . . .	29
7.5	Methods and tools . . . . .	30
7.6	Project phases . . . . .	30
7.7	Conclusions . . . . .	31
7.8	Statistics . . . . .	32
<b>8</b>	<b>Triangulation Games</b>	<b>34</b>
8.1	Overview . . . . .	34
8.2	Project Organization . . . . .	35
8.3	Tools and methods . . . . .	36
8.4	The progress of the project . . . . .	37
8.5	Conclusions . . . . .	37
8.6	Statistics . . . . .	38
<b>9</b>	<b>MetaEdit Team</b>	<b>40</b>
9.1	Overview . . . . .	40
9.2	Organisation and management . . . . .	40
9.3	Methods and tools . . . . .	40
9.4	Project phases . . . . .	41
9.5	Conclusions . . . . .	41
9.6	Statistics . . . . .	41
<b>10</b>	<b>Itku</b>	<b>43</b>
10.1	Yleiskuvaus ohjelmasta . . . . .	43
10.2	Projektiorganisaatio . . . . .	43
10.3	Välineet, menetelmät ja tekniikat . . . . .	43

10.4	Projektin eteneminen . . . . .	44
10.5	Johtopäätökset projektista . . . . .	45
10.6	Mietelauseita . . . . .	45
10.7	Positiivista . . . . .	45
10.8	Negatiivista . . . . .	45
10.9	Tilastot . . . . .	46
<b>11</b>	<b>Webcom</b>	<b>48</b>
11.1	Yleiskuvaus ohjelmasta . . . . .	48
11.2	Projektioorganisaatio . . . . .	48
11.3	Välineet, menetelmät ja tekniikat . . . . .	49
11.4	Projektin eteneminen . . . . .	50
11.5	Projektin ongelmat . . . . .	51
11.6	Johtopäätökset projektista . . . . .	51
11.7	Tilastot . . . . .	51
<b>12</b>	<b>Newsletter Manager Extensions</b>	<b>53</b>
12.1	Projektioorganisaatio . . . . .	53
12.2	Asiakas . . . . .	53
12.3	Projektin aihe . . . . .	53
12.4	Kokemuksia projektin kulusta ja kohdatuista ongelmista . . . . .	54
12.5	Käytetyt työvälineet, työmenetelmät . . . . .	55
12.6	Tilastot . . . . .	56
<b>13</b>	<b>SMS/E-mail yhdyskäytävä</b>	<b>58</b>
13.1	Yleiskuvaus ohjelmasta . . . . .	58
13.2	Projektioorganisaatio . . . . .	58
13.3	Välineet, menetelmät ja tekniikat . . . . .	59
13.4	Projektin eteneminen . . . . .	60
13.5	Johtopäätökset projektista . . . . .	61
13.6	Tilastot . . . . .	62
<b>14</b>	<b>Data visualisation</b>	<b>64</b>
14.1	Yleiskuvaus IntelliToolGraph tuotteesta . . . . .	64
14.2	Organisointi ja hallinta . . . . .	65
14.3	Työkalut ja menetelmät . . . . .	66
14.4	Projektin eteneminen . . . . .	66
14.5	Päätelmät . . . . .	67
14.6	Tilastot . . . . .	67

<b>15 MUPEMAP</b>	<b>69</b>
15.1 Overview . . . . .	69
15.2 Organisation and management . . . . .	69
15.3 Methods and tools . . . . .	69
15.4 Project phases . . . . .	70
15.5 Conclusions . . . . .	71
15.6 Statistics . . . . .	71
<b>16 Korttipaikka</b>	<b>74</b>
16.1 Yleiskuvaus projektista . . . . .	74
16.2 Projektioorganisaatio . . . . .	74
16.3 Välineet, menetelmät ja tekniikat . . . . .	75
16.4 Projektin eteneminen . . . . .	75
16.5 Johtopäätökset projektista . . . . .	76
16.6 Tilastot . . . . .	76
<b>17 Visualize DNA</b>	<b>78</b>
17.1 Overview . . . . .	78
17.2 Organization . . . . .	78
17.3 Problems and risks . . . . .	80
17.4 Management . . . . .	81
17.5 Group meetings . . . . .	81
17.6 Weekly reports . . . . .	82
17.7 Inspections and reviews . . . . .	82
17.8 Methods and Tools . . . . .	82
17.9 Methods and Techniques . . . . .	82
17.10 Conclusions . . . . .	83
17.11 Statistics . . . . .	84
<b>18 CALEX2006</b>	<b>85</b>
18.1 Overview . . . . .	85
18.2 Organisation and management . . . . .	85
18.3 Methods and tools . . . . .	85
18.4 Project phases . . . . .	86
18.5 Conclusions . . . . .	86
18.6 Statistics . . . . .	87
<b>19 Kuvapankki</b>	<b>89</b>
19.1 Yleiskuvaus . . . . .	89
19.2 Projektioorganisaatio . . . . .	89
19.3 Välineet, menetelmät ja tekniikat . . . . .	90

19.4	Projektin eteneminen . . . . .	90
19.5	Johtopäätökset projektista . . . . .	91
19.6	Tilastot . . . . .	91
<b>20</b>	<b>Safety At Work</b>	<b>94</b>
20.1	Yleiskuvaus ohjelmasta . . . . .	94
20.2	Projektiorganisaatio . . . . .	94
20.3	Välineet, menetelmät ja tekniikat . . . . .	94
20.4	Projektin eteneminen . . . . .	95
20.5	Johtopäätökset . . . . .	95
20.6	Tilastot ja kuvia sovelluksesta . . . . .	95



# 1 Usability team

## 1.1 Yleiskuvaus projektista

Lukuvuonna 2005 - 2006 toteutettu projektityökurssi antoi uudenlaisen mahdollisuuden osalle vuorovaikutteisen teknologian opiskelijoista kurssin suorittamiseen. Seitsemän opiskelijaa sijoitettiin kokeiluluontoisesti käytettävyystiimi -nimiseen projektiryhmään, jonka tehtävänä oli vastata muiden projektien käytettävyydestä.

Projektityökurssilla oli kaiken kaikkiaan 20 projektia, joista käytettävyystiimi oli yksi projekti. 19 projektista käytettävyystiimin jäsenet olivat tekemisissä 17 projektin kanssa. Kahdelle projektille ei tarvittu käytettävyysvastaavaa projektityön luonteen vuoksi. Näissä projekteissa ei tehty valmiita käyttöliittymiä, jolloin käytettävyysvastaavan rooli olisi ollut turha. Jokainen käytettävyystiimin jäsen oli vastuussa kahden tai kolmen projektin käytettävyydestä.

## 1.2 Projektiorganisaatio ja hallinta

Käytettävyystiimi toimi itsenäisesti ilman projektijohtajaa koko kurssin ajan ja sen jäseninä olivat: Ivar Ekman, Jaana Huotari, Juha Leimo, Ilari Kajaste, Suvi Peltomäki, Minna Sundström ja Jani Uusiluoto.

Ryhmän sisäinen työnjako mietittiin heti ensimmäisessä palaverissa. Projektit jaettiin vastaamaan jäsenten kiinnostuksen kohteita. Useimmat henkilöt olivat kiinnostuneita samasta projektista, joten alussa projekteilla oli useita käytettävyyseriä edustajia käytössään. Kiinnostuksen kohteiden määrittäessä jokaiselle projektille muodostui oma vastuuhenkilö, joka otti vastuulle osallistumisen vastuullaan olevan projektin viikkopalaveriin, sekä käyttöliittymien ja käytettävyyden suunnitteluun ja joissakin projekteissa jopa toteuttamiseen. Projektien saadessa valmiiksi käyttöliittymiä ja prototyyppisiä tehtiin käytettävyystestejä, joihin pääsääntöisesti osallistui käytettävyyseriä jäsenet. Koko projektin ajan ryhmän jäsenet tekivät keskenään yhteistyötä ja jakoivat tietämystään ja näkemyksiään erilaisista käyttöliittymistä ja ongelmakohtista. Viikkopalaverissa arvioitiin käyttöliittymäsuunnitelmia ja valmiita käyttöliittymiä, sekä tuotiin esille näihin liittyviä ongelmakohtia. Jokaisessa palaverissa valittiin sihteeri ja puheenjohtaja. Ryhmä piti viikkopalaverin kerran viikossa. Projektin alussa ryhmäläiset jakautuivat pienryhmiin, jotka kokoontuivat useammin kuin kerran viikossa. Yhteisten raporttien kirjoittamisessa työnjako jakautui kuka-kuinkin tasaisesti.

### 1.3 Käytetyt työkalut

Käytettävyysryhmällä oli käytössään monia käytettävyydenarvioinnin työkaluja:

Taulukko 1: Projekteihin käytetyt menetit

Metodi	Määrä
Käytettävyysuunnitelma	17
Käyttöliittymäsuunnitelma	9
Käytettävyystestisuunnitelma	3
Käytettävyystestaus	10
Käytettävyystestausraportti	6
Käyttöliittymän walktrough	5
Käyttöliittymän heuristinen evaluointi	6
Muut työt (suunnittelu, protojen teko, käyttöohje, vaatimusmäärittely	16

### 1.4 Projektin vaiheet

Projektin alussa jäsenet valitsivat omat vastuuprojektinsa jonka jälkeen jäsenet aloittivat käytettävyysuunnitelmien teon yhdessä projektien kanssa. Käytettävyysuunnitelmissa määriteltiin käytettävyyksiin rooli ja tehtävät. Tämän jälkeen pyrittiin seuraamaan käytettävyysuunnitelmissa esitettyjä toiveita. Osa projekteista halusi käyttöliittymäsuunnitelman, osa käytettävyystestauksen jo suunnitteluvaiheessa ja osa tarvitsi testauksen vasta soveluksen valmistuttua. Jäsenet osallistuivat myös käyttöliittymän tekemiseen ja muihin työtehtäviin.

Yhteenvedona voidaan todeta, että käytettävyysuunnitelmissa esitetyt työtehtävät ovat toteutuneet hyvin ja olemme säännöllisesti seuranneet työtehtävien etenemistä yhdessä. Jos jokin työvaihe on jäänyt pois tai jos työvaiheita on muutettu suunnitelman jälkeen, on se ollut perusteltua. Kaikki projektit ovat jonkin verran eläneet projektityökurssin edetessä, mikä on luon-

nollista kurssin kokeiluluonteisuuden vuoksi.

Taulukosta 3 näkee projektin jäsenten käyttämät tuntimäärät ja tuotettujen dokumenttien sivumäärät

Taulukko 2: Projekteihin käytetyt tunnit ja sivumäärä per henkilö

Tiimin jäsen	Tunteja	Sivumäärä
<b>Ivar Ekman</b>	<b>252,5</b>	<b>67</b>
<b>Jaana Huotari</b>	<b>232,0</b>	<b>170</b>
<b>Ilari Kajaste</b>	<b>272,0</b>	<b>124</b>
<b>Juha Leino</b>	<b>268,0</b>	<b>243</b>
<b>Suvi Peltomäki</b>	<b>257,0</b>	<b>119</b>
<b>Minna Sundström</b>	<b>251,5</b>	<b>131</b>
<b>Jani Uusiluoto</b>	<b>133</b>	<b>?</b>

## 1.5 Yhteenveto

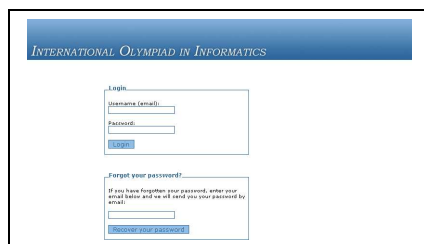
Tänä vuonna projektityökurssissa oli osa vuorovaikutteisen teknologian opiskelijoita sijoitettu omaksi käytettävyyksi, joiden vastuulla oli muiden projektien käytettävyys. Tämä malli vaatii paljon työtä ja sitoutumista sekä vastuunkantoa. Tänä vuonna käytettävyyksiä ei ollut projektinjohtajaa minkä näimme suurena haittana varsinkin alkuvaiheessa kun kokeiluluonteisuuden takia ryhmällä ei ollut valmiita käytäntöjä ja ohjeita toimimiseen. Kokenut projektinjohtaja ja käytettävyyden asiantuntija olisi pystynyt ohjaamaan opiskelijoita paremmin työtehtäviin. Alussa aikaa kului käytäntöjen luomiseen turhaa aikaa. Projektinjohtaja olisi pystynyt paremmin jakamaan työtehtäviä varsinkin kesken projektin, jolloin osalle jäsenistä kasautui monen projektin työt.

Samanlaisella mallilla toteuttaminen vaatii myös paljon uhrauksia, sillä olimme usein vastuullamme olevien projektien aikataulujen varassa. Tosin itsenäinen työskentely antoi myös mahdollisuuden oman työn suunnitteluun. Olimme riippumattomia oman tiimin työskentelyvaiheista, mutta hyvinkin riippuvaisia muiden projektien aikatauluista. Projektien asettamia deadlineja on jouduttu noudattamaan enemmän kuin oman projektin.

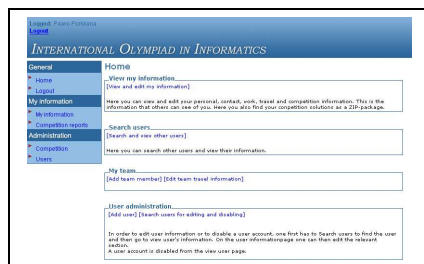
## 2 IOI2005++

### 2.1 Overview

The goal of the IOI2005++ project was to design and implement an application which contains all necessary functionalities needed to administer International Olympiad in Informatics (IOI) competitions. International Olympiad in Informatics (IOI) is an annual competition in computing science for senior pupils at secondary schools all over the world. The homepage of the IOI is located at <http://olympiads.win.tue.nl/ioi/>. The project was a sequel project to the previous year's project IOI2005, which designed the first version of the software. The plan was to re-design the application, evaluate the product of the project IOI2005 and to re-use old usable components and objects and design and implement the missing ones.



Kuva 1: Screenshot of the system's login view.



Kuva 2: Screenshot of the system's main view.

### 2.2 Organisation and management

The client of the IOI2005++ Information System project was Professor Jyrki Nummenmaa, the head of the Department of the Computer Sciences at the

University of Tampere. Professor Nummenmaa is also a member of the IOI Scientific Committee.

The general management of all the project activities was done by the project manager Pauli Borodulin. The project group consisted of five persons:

- Lauri Tuominen
- Jarkko Niemelä
- Satu Mäkitammi
- Terhi Kivinen
- Elina Humaloja

Juha Leino participated in the project on behalf of the Usability Team in co-operation with the project group to design the user interface of the product.

## **2.3 Methods and tools**

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

- Eclipse IDE 3.1.1
- Sun Java 1.4.2
- Web Tools Platform (WTP) 1.0 plug-in for Eclipse
- Concurrent Version System
- Fujaba
- DBSchema plug-in for Fujaba

As almost all of the developers also had full-time job during the project, it was not possible to do teamwork with the group in a designated place. Every developer had the development environment installed on his or her own personal computer and the development was done from home by each developer. The progress was monitored in the weekly meetings and new tasks distributed as previous tasks had been completed.

## 2.4 Conclusions

The project IOI2005++ was a sequel project to the project IOI2005, which created the first version of the software. The plan was to re-implement the application. Most of the design was still accurate, only a few functional features were added like creating and administering events that the competitors can participate during the competitions.

The graphical user interface was completely recreated and most of the navigation was also modified. None of the old code was reusable, so it was not re-used. The database design of the previous project was mostly usable; some tables had to be added, some attributes had to be changed and during the project also some views were also added. The previous project used Fujaba and DbSchema plug-in to handle the data tier. Since some changes to the database were necessary, the Fujaba plug-in had to be used also in this project in order to create the necessary java beans to handle the data. Also the classes that called the java beans that were created by Fujaba had to be rewritten.

The database structure we used was originally created using the same tools as in the previous year project. Only the DBSchema plug-in version we used was newer and due to that, we had to do a conversion for the entire database structure. The conversion process presumed some manual work, but thanks to DBSchema specialist Teppo Lindell at University of Tampere, we managed to do the conversion.

It was quite easy to overcome problems we faced. For example sometimes an attempt to generate \*.java and \*.sql files with the DBSchema plug-in led to a Java exception. A solution was to restart the Fujaba and try the same thing again. Also a database field type selection did not work always as expected. If a field type was set/changed to “auto generated integer”, type “integer” was selected sometimes. However that problem got solved by changing the type to “auto generated integer” once more.

The Fujaba/DBSchema tools did they job at least satisfactory and all of all working with them turned out to be much pleasant experience than originally expected. The user interface was created using Struts framework. The experiences were mostly positive; the most complicated part of the framework was probably maintaining the configuration file, struts-config.xml.

The implementation phase was organized so that four members of the team did Java coding. Each member had Eclipse with Struts features as an integrated development environment on their workstations. The database was used from the server using SSH-tunneling. The source code was updated to the server using CVS.

Though implementation was divided to members according to use-cases, some Java classes (utility) were needed and maintained by multiple people.

CVS was definitely useful tool for the project. Without it, it would have been very difficult for many people to work on the same project at the same time.

## 2.5 Statistics

Taulukko 3: Working hour table of project IOI2005++.

	PREL	REQ	DES	IMPL	TEST	INS,U,MAI	OTHER	Total	%
PLAN	22	34.5	11.5	13.5	2	0	0	83.5	7.9
MEET	53	52	42	64	6	0	0	217	20.4
INSP	9.5	2.5	0	0	0	0	0	12	1.1
STUDY	0	0	0	0	0	0	9	9	0.8
DOCUM	68.5	44	27.5	33.5	16.75	14	2	206.25	19.4
PROD/D	0	13.5	0	0	0	0	0	13.5	1.3
PROJM	13	18	1	0	0	0	1	33	3.1
WORKJ	30	59	116	244	31.5	3	4.5	488	45.9
Total	196	223.5	198	355	56.25	17	16.5	1062.25	
%	18.5	21.0	18.6	33.4	5.3	1.6	1.6		

Taulukko 4: Project's documents.

Document	Version	Pages
Project plan	1.0	46
Project's usability plan	1.0	13
Requirements specification	1.0	38
Implementation plan	1.0	41
User interface document	1.0	49
Test plan	0.1	38
Test report	0.1	6
Maintenance document	1.0	35
Final report	1.0	20
Final story	-	4
Total		290

Taulukko 5: Project's codelines.

Programming language	Java
Total Lines of Code	9333
Method Lines of Code	5751
Number of Static Methods	90
Number of Packages	7
Number of Attributes	263
Number of Interfaces	0
Number of Classes	124

## **3 XML-Search Engine**

### **3.1 Overview**

Purpose of XML-Search Engine project was to upgrade client application of TRIX-search engine server. Old version was made during summer of 2005 and it had performance problems. Also user interface wasn't good enough. More functionality like search results relevance marking was also aim of the project.

### **3.2 Organisation and management**

ex-member: Peng Wang(Coder) ex-member: Saritha Bhoompally(Coder)

The team met every weekly besides the email communicating between the members to discuss the recent achievements and raising problems. The work faced distracting difficulties by the quitting of two members, which happened to be programmers. As result the team had to split the duties of the two missing members on the other remaining members to fill the gap and handle programming and time frame issues.

### **3.3 Methods and tools**

Tools where used: For coding: eclipse. For documentation: MS.office WORD 2003. For UML diagrams: Borland together designer and Rational Rose demo version.

The methods were giving several duties to team members with time frame to move forward to next stage.

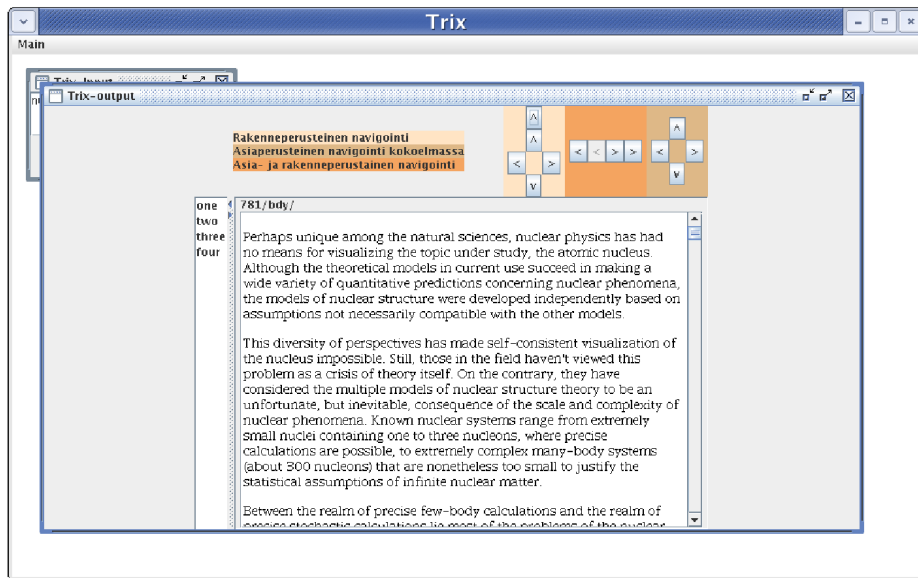
### **3.4 Project phases**

Although the team had a lack of coding experts, the team managed to solve some issues of the prototype such as performance, speed and creating more effective user interface.

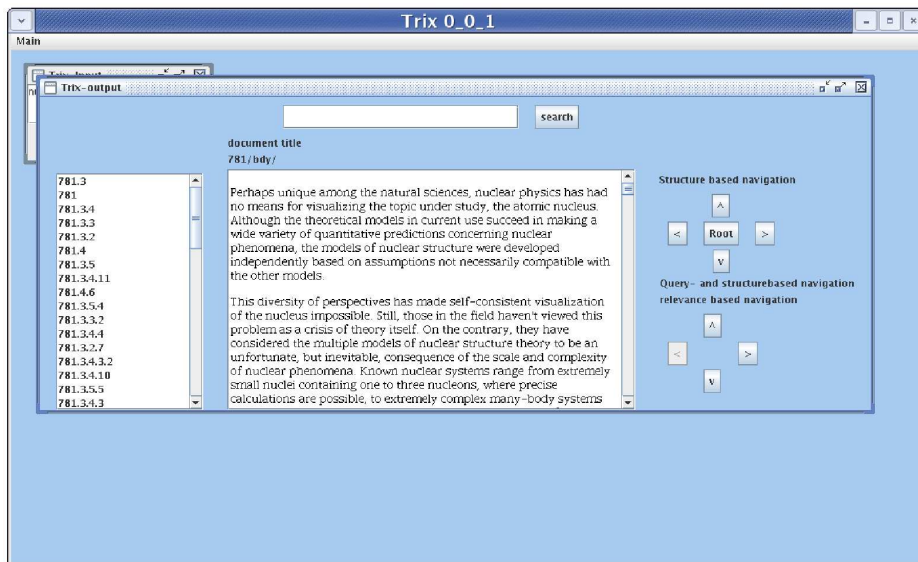
### **3.5 Conclusions**

Of course the course was a great team work experience and quite close to real life. However if we had the chance to go throw it again we would take the probability of members quitting the course more wisely.





Kuva 3: Old user interface of the application.



Kuva 4: New user interface of the application.



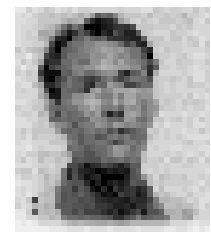
Kuva 5: Team Manager: Marko Elo(Manager and coder).



Kuva 6: Team Member: Sari Miittinen(UI designer).



Kuva 7: Team Member: Sarita Divakaran(Software architecture Designer).



Kuva 8: Team Member: Hannu Nirkkonen(Coder).



Kuva 9: Team Member: Hashem Alsayadi(Tester).

### 3.6 Statistics

Taulukko 6: Working hours table.

	PREL	REQ	DES	IMPL	TEST	INS,U,MAI	OTHER	Total	%
PLAN	1	0	34	4	0	0	2	41	4.64
MEETI	25	31	25	15	11.5	27	14.5	149	16.86
INSP	7	7	9	0	7	0	3	33	3.73
STUDY	8	6	46	7	2	10	7	86	9.73
DOCUM	45.5	5	74	15	38	26.5	21	225	25.45
PRO/D	0	16	46.5	14	5	0	0	81.5	9.22
PROJM	14	8	7	12	4	8	0	53	6.00
WORK	11	18	72	91	8	13.5	2	215.5	24.38
TOTAL	111.5	91	313.5	158	75.5	85	49.5	884	
%	12.61	10.29	35.46	17.87	8.54	9.62	5.60		100

Taulukko 7: Project's documents.

Document	Pages
Project plan	26
Project's usability plan	16
Requirements specification	17
Design plan	29
User interface document	18
Test plan	22
Test report	12
Maintenance document	18
Final report	16
Final story	4
Total	178

Taulukko 8: Old project's codelines.

Ohjelmointikieli	Java
LOC	4317
SLOC	2287
Reused code	
Reused and modified code	
Classes	

Taulukko 9: New project's codelines.

Ohjelmointikieli	Java
LOC	5659
SLOC	2831
Reused code	
Reused and modified code	
Classes	

## 4 Lindelöfin perilliset

### 4.1 Yleistä

Projektin tavoitteena oli luoda tietokanta, johon kerätään tietoja suomalaisista tietojenkäsittelyssä väitelleistä. Tämän lisäksi projektissa toteutetaan käyttöliittymä, jolla voi kätevästi www-sivun kautta etsiä tietoa tietokannasta eri hakuehdoin. Lisäksi ylläpitämisen helpottamiseksi on edelleen kehitetty oma salasanalla suojattu osio sivustossa, jonka kautta tietojen muuttaminen ja päivittäminen sujuu kätevästi.

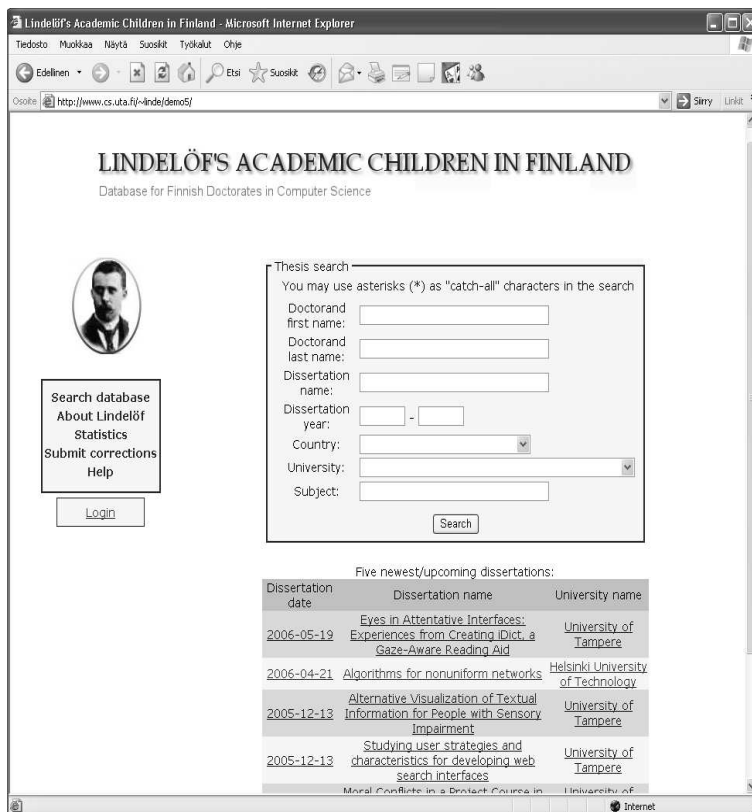
### 4.2 Projektiryhmä ja työjärjestelyt

Projektiryhmämme aloitti syksyllä kuuden hengen kokoisena, mutta varsin pian se supistui viiden hengen ryhmäksi. Rami Törmä toimi projektipäällikkönä. Hän hoiti yleisiä juoksevia asioita ja koordinoi ryhmän tekemisiä. Tanja Lahden vastuulla oli projektisuunnitelman teko yhteistyössä projektipäällikön kanssa ja lisäksi sisällöntuotto tietokantaan. Markus Lervikin aiempi kokemus vastaavanlaisista projekteista oli suureksi hyödyksi. Hänen toimenkuvansa oli ohjelmointi, niin prototyypin kuin varsinaisen tuotteenkin. Sen lisäksi hän osallistui projekti- ja toteutussuunnitelman tekoon. Olli Mäkiketolan rooliin kuului enimmäkseen projektin ulkoasun suunnittelu Sami Salon kanssa. Dokumenteista hän vastasi testaussuunnitelmasta, Loppuraportista ja sen tiivistelmästä. Sami Salo vastasi vaatimusmäärittelystä, projektin kotisivuista ja auttoi sekä Ollia että Tanjaa ulkoasun ja tietojenkeruun kanssa.

### 4.3 Työskentelymetodit ja työkalut

Projektia kehitettäessä käytettiin seuraavia sovelluksia ja työvälineitä:

- Creole/Propel (viimeisin SVN-versio 2006-01-17)
- PHP (Versio 5.1.2)
- PostgreSQL (8.1)
- Ohjelmointityökaluna (IDE) Eclipse
- Xored TruStudio (1.0.1)
- EMS SQL Manager 2005 for PostgreSQL Lite (Windows, versio 3.4.0.4)



Kuva 10: Main screen.

Eclipse ja TruStudio ovat olleet loistava apu koodaamisessa. Intellisense ja ”code completion” helpottaa ja nopeuttaa koodaamista todella paljon.

Creole ja Propel ovat yhtäläillä olleet tärkeitä työkaluja. Propel tekee tietokannan käsittelystä PHP:llä helppoa tarjoamalla relaatiotietokannalle olioabstraktion. Arvioimme sen vähentäväneen koodirivien määrää noin kolmanneksella ja työmäärää ehkä jopa noin 60% projektissamme. Creole puolestaan mahdollistaa myöhemmin pinnan alla olevan tietokannan vaihtamisen helposti.

EMS on helpottanut tietojen lisäystä ja muutenkin tietokannan rakenteen tekemistä. Se ei ole ollut välttämätön mutta kiva apu kuitenkin. Sen suurin hyöty on näkynyt tietokannan luonnin nopeutumisessa.

Projektimme ajautui luonnollisesti siihen ratkaisuun, että työskentelimme enimmäkseen kukin omilla tahoillamme. Tällöin palavereista tuli sitäkin tärkeämpiä kokoontumishetkiä, kun saattoi taas kasvokkain tavatessa puida omia ongelmakohtiaan muiden kanssa.

## 4.4 Projektin vaiheet

Projekti aloitettiin rauhallisesti syksyllä, selvittäen mistä projektissamme oikein on kyse. Vaatimusmäärittelypalavereissa selvisikin, että kyseessä oli aika inhimillisen kokoinen projekti meidän kuuden hengen innokkaalle ryhmällemme. Kuitenkin vaatimusmäärittelydokumentin kirjoituksen aikoihin paljastui, etteivät kaikki ryhmäläisemme olleet yhtä innokkaita. Tässä vaiheessa ryhmän yksi jäsen jätti koko touhun sikseen vieläpä kaikkein ikävimmällä tavalla. Hänen tehtävänsä oli kirjoittaa vaatimusmäärittelyä, jonka deadline uhkaavasti lähestyi. Silti hän ei turhaan vaivautunut mainitsemaan kurssin kesken jättämisestä millään lailla meille tai luennoitsijalle. Tästä takaiskusta kuitenkin toivuttiin ja pikaisesti jakamalla hänen osuuttaan kirjoitusurakasta muille, saatiin ensimmäinen suurempi takaisku kunnialla hoidettua.

Joulun aikoihin alkoi olla melko kiireistä, kun prototyyppi piti saada aikaiseksi ja valmistautua esitykseen. Proto saatiin kuitenkin riittävän valmiiksi, että sitä kehtasi esitellä ja esitysikin sujui hyvin. Projektiryhmä pääsi näin ansaitulle joululomalle. Kevätkaudella päästiin tositoimiin, kun enimmän ajan saattoi laittaa tulevan lopullisen tuotoksen tekemiseen. Muutamia odottamattomia takaiskuja tuli projektin ollessa kiivaimmillaan. Kehityskoneella, jossa pyöri Apache ja sovelluksen tuorein versio, esiintyi ihmeellisiä virheilmoituksia joita ei saatu paikallistettua kovinkaan nopeasti. Ongelmaksi paljastui tässä tapauksessa Skype-nettikeskusteluohjelma. Se aiheutti ongelmia Apachen kanssa käyttäen porttia 80 omiin tarkoituksiinsa. Toinen odottamaton ongelmanaiheuttaja oli yliopiston Horde-sähköpostijärjestelmä, joka onnistui rikkomaan liitetiedostot.

Ihan viimeisinä viikkoina kiirettä on pitänyt, kun tammikuussa ei otettu riittävän vakavasti sitä ajantarvetta, jonka projekti vaatii. Lopputulos on kuitenkin hyvä ja pitämällä kiirettä ollaan tuote saatu jalostettua paremmaksi kuin oltaisiin osattu kuvitella. Varsinkin ulkoasun suhteen oltiin skeptisiä, kun kukaan ei tunnustanut aluetta vahvimaksi osaamisalueekseen. Onneksi epäilyt olivat turhia!

## 4.5 Yhteenveto

Projektiryhmän mielestä projekti ja kurssi onnistuivat hyvin. Kovinkaan monessa tietojenkäsittelyn kurssissa ei pääse kokemaan samanlaista pitempijaksoista isomman ryhmän yhteistyöskentelyä. Lopputuotekin näyttää hyvältä ja täyttää vaatimukset, jotka sille asetettiin. Huomasimme kuitenkin projektia työstäessä, ettemme asettaneet itsellemme riittävän ajoissa olevia tarkistustappoja projektin eri vaiheiden etenemisen suhteen. Muutamaaan kertaan tämän takia projekti ei pysynytkään ihan aikataulussa, vaan loppua kohti etapit tahtoivat päästä luisumaan käsistä.

## 4.6 Tilastoja

Taulukko 10: Työtunnit projektille Lindelöfin perilliset.

	Esitutk	Vaat	Suun	Toteu	Test	AS,KO,YP	Muu	Yht.	%
TUMI	5	8,25	9	12,5				34,75	7,0
PALA	8,25	13	26,75	26	7			81	16,2
TARK		11	1,75	5	1			18,75	3,8
OPET	4,8	2	0,5	2,75	1,5		11,25	22,8	4,6
DOKU	0,5	3,5	33,75	19	24,75	8	11	100,5	20,1
PRO/D		7	9,25					16,25	3,3
PROJH	4	13,75	4,25	8	5,75			35,75	7,1
TYÖ	0,5	12,5	30	123,5	20,75		2	189,25	37,9
Yht.	23,05	71	115,25	196,75	60,75	8	24,25	499,05	
%	4,6	14,2	23,1	39,4	12,2	1,6	4,9		



Taulukko 11: Projektin dokumentit.

Dokumentti	Sivua
Projektisuunnitelma	23
Käytettävyysuunnitelma	6
Vaatimusmäärittely	26
Toteutussuunnitelma	9
Käyttöliittymäsuunnitelma	6
Testaussuunnitelma	15
Testausraportti	3
Asennusdokumentti	3
Ylläpitodokumentti	8
Loppuraportti	14
Loppuraportin tiivistelmä	3
Yhteensä	116

Taulukko 12: Projektissa kirjoitetut ja generoidut koodirivit.

Ohjelmointikieli	PHP
Rivejä (LOC)	5298
Rivejä ilman kommentteja ja tyhjiä rivejä (SLOC)	4287
Creole + Propel + generoitu koodi	31888

## 5 SysMLL

### 5.1 Overview

### 5.2 Organisation and management

The MUPEMAP team consists of the following members and speciality areas:

- Teemu Mäki, team lead
- Kamrul Ahsan, databases and PHP coding
- Thanyaporn Lerlerdthaiyanupap, PHP coding
- Henri Rantala, PHP coding, usability
- Paavo Toivanen, PHP coding, browser compatibility

Kamrul Ahsan also belonged to the requirements engineering group, along with Catalin Ionescu, Domingo Diez Barrero and Sun ”John” Jiapu. John was also part of the actual project group in the beginning, but dropped out later.

The project group had regular meetings on almost once a week. The client participated very actively in the team’s meeting (present in nearly every meeting). The project members reported their hours to the project manager using a semi-automatic web-based interface.

### 5.3 Methods and tools

- hamppu.uta.fi used as the developing platform. Some problems due to the way the server was configured (error messages and access limitations), also the user accounts were delayed pretty long.
- No special development tools were used.
- The project had a CVS repository, but it was not effectively used.
- Moodle used as an online collaboration tool and document storage. Worked fine for online conversations, but some drawbacks with document management, e.g. limited attachment size, editing the entries limited to 15 minutes after initial posting time, etc.
- A web-based tool for reporting hours to the project manager. The project manager extracted the data manually, but the tool ensured all reports could easily be found in one place.

## 5.4 Project phases

In the requirements elicitation phase, the work proceeded rather smoothly and in schedule thanks to active participation from the client. Because the requirements engineering group consisted partly of same members as the project team itself, some information sharing was spared. However, the team found that the specifications made by the RE team were unrealistic and unfitting and the database structure was not optimal. Much work was needed to make it acceptable. A simple UI prototype was made in time for the autumn presentation. Schedule slippage began after Christmas holiday, as there were some difficulties fitting the different time schedules together again. Also, the problem setting stayed unclear and the requirements were changed many times even during test plan writing and implementation writing stage. Problems with Unicode encoding and browser compatibility caused some further slippage when the coding began. To counter the delay, extensive testing was skipped. The product was about 90% finished by the end of the course, but due to the client's request, the release was extended to the end of May.

## 5.5 Conclusions

The project was an overall success. The client's active participation made it easy to perfect the design. However, the waterfall development model seemed not to be optimal for this kind of interaction with the client. Incremental or Extreme Programming approached could have worked more effectively. One strength of this project team was that all members were able to participate in the coding.

Some critique about the weekly meetings:

- Usually no clear agenda for each meeting, which allowed the client to run the conversation
- Many absences from the weekly meeting (the meeting times were regular so this shouldn't have been a problem)
- Deadlines discussed too late - often no clear visibility to the overall schedule

Miscellaneous critique:

- Project webpages were left unattended, as no-one was assigned responsibility for them.

- The workload was unbalanced between people and different stages of the project. The autumn was dominated by redundant planning and meetings with little output.
- Too little personal guidance from the team leader in some issues.

## 5.6 Statistics

Taulukko 13: Working hour table of project SysMLL.

	PREL	REQ	DES	IMPL	TEST	Total	%
Planning	11	27	13	27	0	77	6,3 %
Meetings	48	28	16	62	0	154	12,6 %
Inspections	0	3	6	6	0	15	1,2 %
Studying	57	39	16	61	0	277	22,8 %
Documenting	6	23	23	37	0	89	7,3 %
Prototype / demo	0	2	21	6	0	29	2,4 %
Project mgmt	8	13	15	24	0	60	4,9 %
Work	3	6	48	458	0	515	42,3 %
TOTAL	125	132	157	681	0	1216	100,0

Taulukko 14: Project's documents.

Document	Pages
Project plan	30 pages
Project's usability plan	TBD
Requirements specification	TBD
Implementation plan	52 pages
Test plan	TBD
Test report	TBD
Maintenance document	TBD
Final report	TBD
Final story	4 pages
Total	TBD

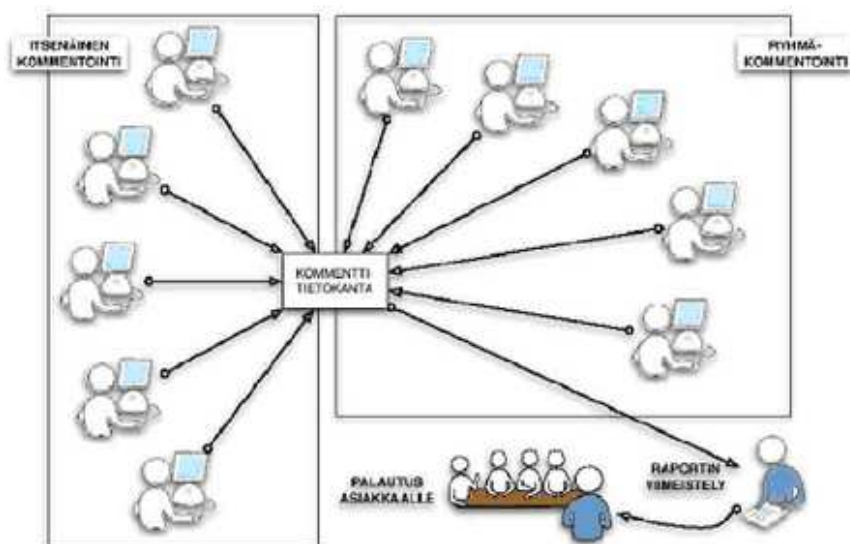
Taulukko 15: Project's codelines.

Language	PHP
LOC	3600 LOC
SLOC	3000 LOC
Reused code	N/A
Reused and modified code	100 LOC
PHP functions	20
Language	CSS
LOC	260 LOC
SLOC	260 SLOC
Reused code	N/A
Reused and modified code	30 LOC
Language	JavaScript
LOC	600 LOC
SLOC	560 SLOC
Reused code	100 LOC
Reused and modified code	50 LOC
Language	PostgreSQL
LOC	150 LOC
SLOC	N/A
Reused code	N/A
Reused and modified code	N/A

## 6 Heuristisen arvioinnin tukijärjestelmä

### 6.1 Yleiskuvaus ohjelmasta

Projektin tavoitteena oli toteuttaa heuristisen arvioinnin tukijärjestelmä Tampereen yliopiston käytettävyysslaboratoriolle. Järjestelmällä voidaan taltioida, kommentoida ja raportoida heuristisessa arvioinnissa saatuja tuloksia. Koska järjestelmä oli tarkoitus toteuttaa käytettävyyssarvioinnin ammattilaisille, myös järjestelmän käyttöliittymän tuli noudattaa niitä arvoja, joidenka pohjalta kyseiset asiantuntijat työssään testaavat järjestelmiä. Tästä syystä käytettävissämme ollut käytettävyyssryhmän edustaja keskittyi pääasiassa ainoastaan meidän projektiimme. Lopputuotteesta tuli WWW-pohjainen verkkosovellus, jota useammat ihmiset pystyvät samanaikaisesti käyttämään.



Kuva 11: Arviointiprosessin kulku.

Kuvassa 11 on esitetty tietokoneavusteisen heuristisen arviointiprosessin kulku. Prosessi on jaettu eri vaiheisiin.

- Itsenäisen kommentoinnin tilassa arvioijat voivat tuottaa järjestelmään kommentteja ja muokata omia kommenttejaan. Arvioijat eivät kuitenkaan voi nähdä toistensa kommentteja.
- Ryhmäkommentointitilassa arvioijat näkevät toistensa kommentit ja voivat kommentoida näitä.

- Raportointitilassa arvioinnin raportoija voi muokata muiden kommentteja ja generoida lopuksi järjestelmän kommentteista käytettävyyssraportin.

## 6.2 Projektioorganisaatio

Kuten yllä mainittiin, projektin asiakkaana oli Tampereen yliopiston käytettävyysslaboratorio. Käytettävyysslaboratorion edustajina toimivat alla olevat henkilöt.

- Saila Ovaska
- Harri Siirtola

Projektiryhmän projektipäällikönä toimi Pauli Borodulin. Projektipäällikön lisäksi ryhmään kuului alla olevat henkilöt.

- Matias Muhonen
- Tuomas Tauriala
- Timo Klemetti
- Aaro Tuomisto
- Tuukka Pasanen

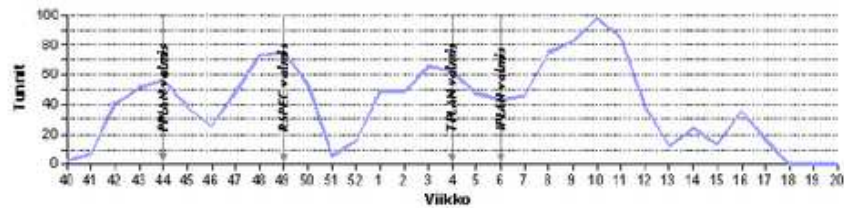
Käytettävyyssryhmän edustajana toimi Ilari Kajaste.

## 6.3 Projektin eteneminen

Projektin alussa ryhmällä ei ollut kurssin suorittamisen kannalta erityisen korkeita tavoitteita. Kuitenkin kurssin edetessä motivaatio kasvoi huomattavasti hyvän ryhmähengen ja mielenkiintoisen aihealueen ansiosta.

Kurssivaatimuksiin kuuluvien dokumenttien parissa tuli pakerrettua useita tunteja. Tarkalleen dokumentteja tehtiin noin 18 prosenttia koko projektiin varatusta ajasta. Kuvasta 12 voi nähdä miten projektiryhmän tunnit jakautuivat eri dokumenttien kesken.

Järjestelmän arkkitehtuuriratkaisuun vaikuttivat järjestelmän geneerisyys, laajennettavuus sekä ylläpidettävyys. Projektin alusta lähtien tiedettiin, että kaikkia vaatimuksia ei projektille varatun työmäärän puitteissa voida toteuttaa. Tästä syystä arkkitehtuuria suunniteltaessa pyrittiin mahdollistamaan toimintojen helppo lisääminen mahdollisissa jatkoprojekteissa.



Kuva 12: Työmäärien jakauma.

Järjestelmä toteutettiin lähes kokonaisuudessaan viiden viikon aikana. Tänä aikana järjestimme useita workshop-tilaisuuksia, joissa istuimme kaikki yhdessä toteuttamassa järjestelmää.

Vesiputousmallin mukainen projektin organisointi oli liian raskas ottaen huomioon käytettävissä olevat resurssit. Esimerkiksi inkrementaalinen ohjelmistokehitys olisi tehostanut ohjelmiston kehittämistä huomattavasti.

## 6.4 Projektin hallinta

### Palaverit

Projektiryhmä kokoontui kerran viikossa. Tapaamisissa seurattiin projektin etenemistä, tai työskenneltiin workshop-tyyppisesti. Lisäksi asiakkaan kanssa järjestettiin tapaamisia tarpeen mukaan.

### Viikkoraportit

Projektipäällikkö seurasi tuntimääriä viikkotasolla. Kukin projektin jäsen raportoi työtuntinsa viikon päätteeksi.

### Tarkastukset ja katselmoinnit

Projektin tärkeimmät dokumentit katselmoitiin yhdessä asiakkaan ja ohjausryhmän kanssa. Tämän lisäksi projektiryhmä katselmoi myös sisäisesti osan dokumenteista ennen niiden julkaisua varmistaakseen dokumenttien hyvän laadun.

### Muut

Projektin hallinnoinnin ja kommunikoinnin apuna käytettiin sähköpostilistaa.



## 6.5 Käytetyt välineet ja menetelmät

Projektin jäsenet käyttivät työvälineenä Eclipse-sovelluskehittäjä ja Subversion-versionhallintaa. Projektiryhmä hyötyi näistä työvälineistä, ja ne lisäsivät työskentelymukavuutta. Varsinkin versionhallintaohjelma koettiin tarpeelliseksi, sillä yhteisiä tapaamisia ei pystytty järjestämään usein.

Ohjelmiston toteutuksessa hyödynnettiin Spring-sovelluskehystä ja Hibernate-kirjastoa. Ohjelmistojen versiot löytyvät projektisuunnitelmasta.

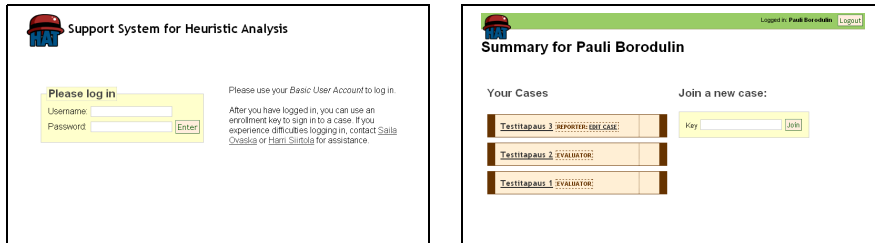
## 6.6 Ongelmat

Taulukko 16: Projektin aikana toteutuneet ennakoitujen riskit.

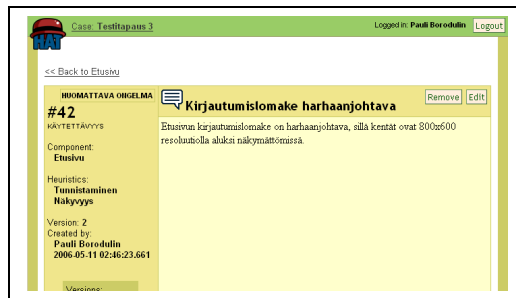
Riski	Syy	Seuraukset	Toimenpiteet
Työkalujen uutuus projektiryhmän jäsenille	Osaamistaso ja puutteellinen koulutus	Toteuttamiseen käytettävä aika väheni	Ryhmän sisäinen koulutus
Ryhmän jäsenten aikataulujen tiukkuus	Muut opinnot ja työssäkäynti	Projektin käytettävissä olevat resurssit vähenivät	Pyrittiin jakottamaan tehtäviä aikataulujen sallimissa puitteissa
Asiakkaan kyky määrittellä tuotteelta vaadittavat ominaisuudet	Asiakkaan riittämätön paneutuminen ja esisuunnittelu	Muutospaineet projektin loppuvaiheessa	Muutospyyntö priorisoitiin ja toteutettiin mahdollisuuksien rajoissa
Projektin resurssit loppuvat kesken	Kurssin opintoviikkomäärän asettama tuntirajoitus	Projektiryhmä ei ehtinyt toteuttaa kaikkia toimintoja	Toteutettiin vain kriittisiksi luokitellut toiminnot

Etätyöskentely vaikeutti kommunikaatiota ryhmän jäsenten välillä, ja siten mahdollisesti vähensi projektin käytössä olevia resursseja. Ongelmaa yritettiin hoitaa parantamalla kommunikaatiota IP-puheluilla.

## 6.7 Ruutukaappauksia järjestelmän näkymistä



Kuva 13: Kuvakaappaukset järjestelmän kirjautumis- ja päänäkymästä.



Kuva 14: Kuvakaappaus arviointitapauksen kommentinäkymästä.

## 6.8 Tilastot

Taulukko 17: Projektin HAT tuntitaulukko.

	Esit	Määr	Suun	Tot	Test	As, Ko, Yp	Muu	Yht	%
Tumi	11	37,5	41	29,5	0	0	0	119	8,7
Pala	67	69,5	58	47,5	0	10	0	252	18,4
Tark	9,5	5,5	15,5	0	0	0	0	30,5	2,2
Opet	0	0	0	0	0	0	19	19	1,4
Doku	47,5	61	91	7	11	18	0	235,5	17,2
Pro/D	0	24,5	5	0	0	0	4	33,5	2,5
Projh	12	14	0	1	0	0	0	27	2,0
Työ	37	78	74	409	41,5	2,5	7	649	47,6
Yht	184	290	284,5	494	52,5	30,5	30	1365,5	
%	13,5	21,2	20,8	36,2	3,8	2,2	2,3		

Taulukko 18: Projektin dokumentit.

Dokumentti	Versio	Sivuja
Projektisuunnitelma	1.1	42
Vaatimusmäärittely	1.4	47
Käytettävyyssuunnitelma	1.0	13
Käyttötapaukset	1.1	16
Testaussuunnitelma	1.1	47
Toteutussuunnitelma	1.2	34
Käyttöliittymäsuunnitelma	1.0	48
Käyttöliittymän näyttökartta	1.0	3
Käyttöliittymän näkymähierarkiakaavio	1.0	3
Käyttöliittymän käyttöliittymänavigointikaavio	1.4	3
Testausraportti	1.2	10
Ylläpito-ohje	1.1	29
Kehitysympäristön asennusohje	0.1	22
Loppuraportti	0.4	22
Loppuraportin tiivistelmä	-	5
Yhteensä		344

## 6.9 Koodin määrä

Seuraava tilasto on tuotettu projektin lähdekoodeista käyttäen Eclipse-laajennosta Metrics (<http://metrics.sourceforge.net>). Käytetty laajennos osaa laskea automatisoidusti annetusta Eclipse-projektista joukon erilaisia metriikoita, joista alle on liitetty osa.

Taulukko 19: Koodirivien lukumäärä.

Ohjelmointikieli	Java
Koodirivejä	5945
Metodien koodirivejä	3018
Staattisia metodeja	10
Pakkauksia	13
Attribuutteja	227
Rajapintoja	40
Luokkia	105

## 7 DivXML Editor



### 7.1 Overview

This product provides the functions helping XML users be able to generate a graphical user interface (GUI), directly from the XML schema, which will help the user to enter the data, and in result a well-formed XML document will be generated. The users will also be able to load the existing well-formed XML documents and the application will check whether they are valid or not.

### 7.2 Organisation

#### Project Group

At the beginning of the project, project group had 6 members including project manager. Situation changed at the beginning of the year 2006 and currently we have 5 members and no project manager. Current members are: Ahmer Iqbal, Tayyab Zaheer, Wenfeng Liu, Juuso Näsi and Jari Kivelä.

#### Client

Client of the project is University of Tampere, department of computer sciences, TAUCHI unit. Contact person for this project is Ivan Tugoy.

## **Others**

Usability supporting team (U-team) and especially their contact person Ivar Ekman has been integral part of the project team giving support on usability issues and also giving new ideas on how to visualize XML structure.

## **7.3 Problems and risks**

### **Foreseen risks**

Project plan describes several possible risks for the project. There were for example technology risks since XML wasn't that well known to the group members and especially managing the XML tree and using schema proved to be difficult tasks. Also human risks actualized themselves since project manager decided to leave at the beginning of the year 2006 due to personal reasons. Our group was multinational and especially at the beginning there was a language problem and we had to be very strict with the terminology so that there wouldn't be any misinterpretations. This eased towards the end since everybody knew each other much better and discussion flowed more freely.

### **Risks not foreseen**

There weren't risks that we hadn't at least somehow anticipated. What was unforeseen was the scale how some of the actualized risks affected our project. We did mention the risk of losing a member or even project manager, but we didn't know how drastically it really affected the project. Also some of the techniques used in programming appeared to be far more difficult and taking much more time than we anticipated. For example schema was a really hard and difficult part of the program and we didn't foresee the scale of that problem at the beginning of the project.

## **7.4 Management**

### **Group meetings**

Group meetings were held usually once a week and in some phases once in two weeks or twice a week depending of the project status and issues at hand. It was sometimes difficult to get all the members to attend the meetings at the same time, since two of the members had full time jobs and other members had many other courses in addition to project course. When we eventually got in the same place at the same time, usually huge progress was made through brainstorming.

## **Weekly reports**

Weekly reports were sent at the beginning of the project by the project manager and later by group members. Due to missing project manager, there were some misunderstandings on duties concerning project communication to all the necessary stakeholders.

## **7.5 Methods and tools**

The following tools were used during the project.

- Working hour tracking system (Electronic Task/Time Recording, ETTR 1.0b)
- Java Development Environment with NetBeans IDE
- CVS versioning system

The CVS system was only partly used since there were technical difficulties in implementation. When CVS failed to perform acceptably, normal e-mail was used for both code and documentation distribution between members.

The working hour tracking system was used in order to keep track of project members' working hours as well as to keep control over the tasks accomplished. The tasks were entered into the system by project members themselves. This was also the basis for all working hour reports.

Some extreme programming has been taking place during the development. This was considered very helpful when difficult development issues were at hand.

## **7.6 Project phases**

The development process was broken down into the following phases. Each of the phases was divided into task areas.

### **Preliminary Analysis**

- Preliminary study on the subject

### **Requirements Engineering**

- Revision of the requirements specification as written by the RE group
- Designing a paper prototype
- Research on existing solutions

## **Design**

- Writing a test plan (according to the requirements specification)
- 'Proof of Concept' (POC) programming
- Designing the application and writing the design document
- Developing a prototype (using POC program parts) and testing it with users who are inexperienced regarding XML

## **Implementation**

- Planning the Implementation (and writing the respective document)
- Implementing the application (and testing written single parts)
- Writing the user manual
- Final test (integration and usability test) according to requirements specification and usability guidelines

## **Testing**

- All delivered artifacts (including the application) are tested

## **Installation and project ending**

- Installation to client
- Project presentation and finalizing
- Final documentation

When project moved to incremental software development, phases 'Design', 'Implementation' and 'Testing' were repeating in cycles so that the project had some kind of version of the product at hand at all times.

## **7.7 Conclusions**

### **Experiences from the project**

Generally the project was described as hard by the project members. So many of the risks actualized and they made almost crucial blow to the member motivation. Also at the beginning project members weren't really aware of the client requirements and we experienced a 'moving target' effect. This was due to miscommunication since after the client started to involve more with

the project group the requirements became much clearer to every member. Usually when project manager leaves, project comes to a state of 'no steering, no advancement'. But even after this setback the project continued to move on, mainly because our client became almost like one of the project members. High client presence at everyday project work was perhaps unorthodox but was considered by every member as highly valuable. The prioritization of the requirements based on mandatory and non-mandatory requirements, the high client presence in every meeting, and technical support for confusing things made this project finally work.

### **What to do better next time**

Communication was not optimal at some points and therefore we had problems knowing what everyone was doing. This could have avoided, at least partly, by making sure that everyone has time for the meetings. At the beginning we had fixed meeting time, but since everyone had many other duties, it wasn't efficient if we really wanted to have everyone at the meeting. Finally we moved to 'election' type meeting arrangement where the best possible timeslot for the meeting was selected using e-mail voting. This proved to be very good idea for these kind of projects where everyone has many other responsibilities in their lives too. Also since project manager is so crucial to the project, we should have asked more help from the course teachers. We did move to the incremental development model, but it wasn't sure if it really helped the project since some of the tasks were crossing over the development cycles anyway because of technical problems. Trying to keep up with the cycles might even have given more overhead. Trying to focus to main features appeared to be better decision as suggested by the client. Keeping the focus is a difficult task and should be done by the project manager, in our project the focus was fortunately kept by our client.

## **7.8 Statistics**

Statistics until 17th of May. Please note that hours are subject to change.



Taulukko 20: Working hour table of project DivXML Editor.

	PREL	REQ	DES	IMPL	TEST	INS,U,MAI	OTHER	Total	%
PLAN	22,3	3,3	10,0					35,6	2,7
MEET	8,3	91,7	58,3	88,7	4,2			251,2	18,8
INSP		28,8	2,5		1,0			32,3	2,4
STUDY	9,5	20,1	37,7	66			1,5	134,8	10,1
DOCUM	9,1	29,9	15,8	36,2			7,0	98	7,3
PROD/D		0,3	5,5	17,3				23,1	1,7
PROJM	7,2	13,8	1,8	18				40,8	3,1
WORKJ				510,2	207,7			717,9	53,8
Total	56,4	187,9	131,6	736,4	212,9	0	8,5	1333,7	100
%	4,2	14,1	9,9	55,2	16	0	0,6		

Taulukko 21: Project's documents.

Document	Pages
Project plan	31
Requirements specification	20
Test plan	20
Implementation plan	10
Test report	5
Final report	10
Final story	5
Maintenance document	15
User's guide	10
Total	126

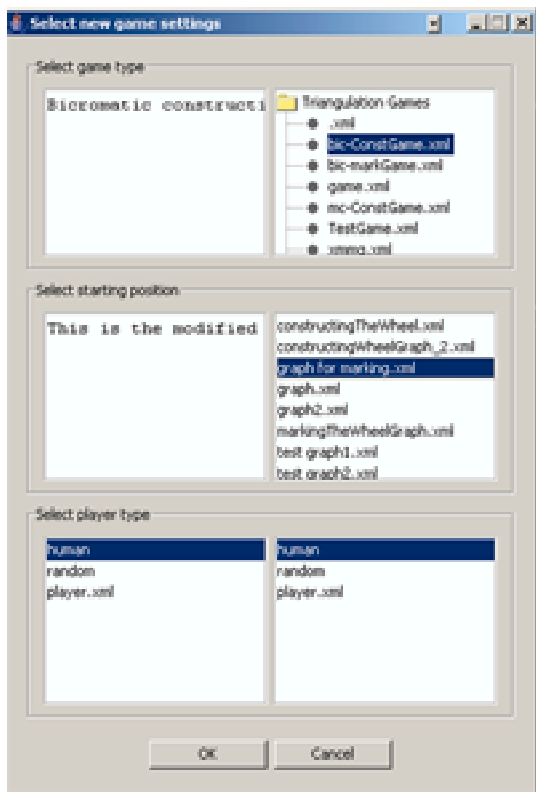
Taulukko 22: Project's codelines.

Programming language	Java
LOC	2900
SLOC	1700
Reused code	500
Reused and modified code	1100
Classes	300

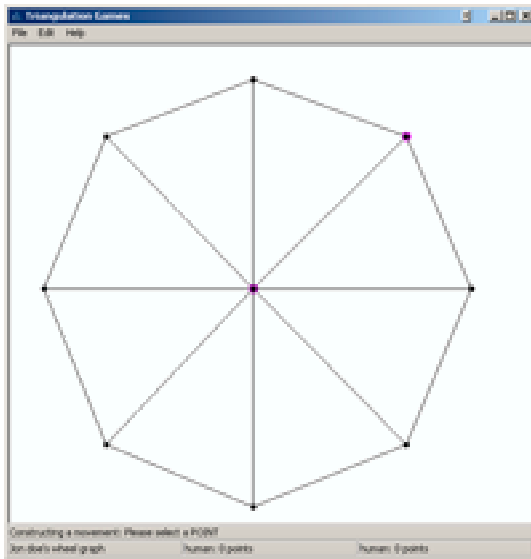
## 8 Triangulation Games

### 8.1 Overview

The goal of the Triangulation Games project is to create standalone Java 2 software for playing and editing triangulation games explained in an article by Aichholzer et al. (Games on triangulations, *Theoretical Computer Science*, 343 (2005), 47–71). The program will be released under GPL-license which means that the program and it's source code will be available for anyone. The program is mainly made for scientist who are interested in studying game algorithms and artificial intelligence, but also for common people who like to play triangulation games.



Kuva 15: The game selection view.



Kuva 16: The game view.

## 8.2 Project Organization

All the members of the project group are studying Information Sciences in the University of Tampere. For the group, the triangulation games project is a part of our studies.

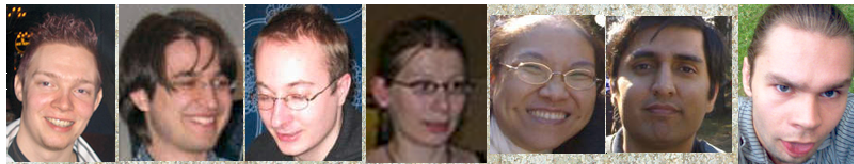
The project manager is Ville Parviainen. He has been in charge of the overall progress of the project, as well as handling the different relations between the project group and the client. The manager handled the requirements of his task exceedingly well, and took advantage of each members' skills and knowledge efficiently. The interaction between the manager and the group, and also within the whole group was solid and worked great lengths towards helping the whole project become a success.

The members of the group include people from different countries and backgrounds. Proceeding alphabetically, the members and their respective responsibilities are listed in the following:

- Kyösti Karila was in charge of the finalising of the requirements specification document. He also worked with implementing the XML -file system into the project.
- Umair Khan was responsible for compiling the original project plan document. Umair's main responsibility within the actual implementation was the graphical user interface.

- Suvi Peltomäki was the usability team contact person, and designed the prototype for the GUI and the usability plan document, amongst other things.
- Salvador Romero constructed the implementation plan document, and his role was essential in many parts of the actual implementation.
- Jon Sahlberg also participated in completing the implementation plan document, as well as handling the implementation of the artificial intelligences and the actual games.
- Chienting Weng was in charge of the test plan. She also had the responsibility of integrating the XML -file reader into the actual program.

In addition to the named responsibilities, each member of the group participated in the testing process.



Kuva 17: Project group.

### 8.3 Tools and methods

The of tools and different working methods used in the project is as follows:

- Java 2 Platform, Standard Edition (J2SE). During the development of the project we have been using versions 1.4 and 1.5 interchangeably. This enables the Triangulation Games software to work in any computer which has Java 2 (version 1.4 or later) Runtime Environment (JRE) installed. Using the Java platform for this project enables running the application on multiple different environments without any modification to the code. This makes it available for as many developers or other interested parties as possible.
- Eclipse SDK version 3.0. This programming environment has been extremely effective and useful. With the help of Eclipse, the actual programming of the project has been well-organised and understandable, especially with many different members of the group programming the game simultaneously.

- CVS. The use of the concurrent version surveillance has been equally helpful in the simultaneous programming. Having been able to browse through the different changes and modifications in the code has helped each member of the group to understand and visualize the nature of the program.

## 8.4 The progress of the project

The progress of the project was conducted under an efficient, yet flexible time table. The project manager had planned each phase of the project thoroughly, and the group only had to follow the premeditated timing to stay within the bounds of the schedule. The first part was to get acquainted with the concept of triangulation games. After studying the logic behind these kinds of games, the group could concentrate on thinking of different ways to plan and implement the project.

The first couple of months were spent on intense planning and discussing various ideas. After coming up with the necessary documents, the group could then proceed to aim its energy towards the actual implementation of the project. This was understandably the most time-consuming part of the project. Along the way, there were times when the workload seemed to be a bit much, as the group also had other studies and such to attend. But at the end of the day, the group is satisfied with the progress and the results of the project. The project has also managed to stay on course as far as the timing and deadlines, at least for the most part.

## 8.5 Conclusions

The overall feeling amongst the group is very positive. Being part of this project has been extremely useful as far as our education, even if its contents are not completely compatible with each of the members' studies. Working very tightly as a group has enforced our interaction skills, and has served as excellent practise for the upcoming challenges that we may encounter.

Furthermore, it has to be stated that because of the very international structure of the group, the project also has served as a means of cultural education. The members of the group are now more familiar with people from other parts of the world, which can only be viewed as a very meaningful and important characteristic of the project work course.

It is obvious that there have been some things that could have been conducted differently along the way. But these have mainly been very indifferent by nature, and ultimately the group will view the whole project as a success.

## 8.6 Statistics

Triangulation games Update: 8.5.2006  
Working hour report

	PRELAN	REQUIRE	DESIGN	IMPLEM	TEST	INS,U,MAI	OTHER	TOTAL
PLANNING	6,00	9,00	10,00	4,00				29,00
MEETINGS	42,50	22,50	33,50	55,50	8,50		8,00	170,50
REVIEWS	21,00	5,50	2,00	8,00	2,00			38,50
STUDYING	70,00	18,50	55,00	36,00	5,00		2,00	186,50
DOCUMENTING	66,00	42,00	26,50	12,00	8,00		1,00	165,50
PROTOTYPE DEMO		2,25	3,00	26,00	6,00			47,25
PROJECT MANAGEMENT	16,00	12,00	8,00	25,00	6,00	3,00	6,00	76,00
WORK (the real work)			5,00	221,00				226,00
<b>TOTAL</b>	<b>221,50</b>	<b>111,75</b>	<b>143,00</b>	<b>397,50</b>	<b>36,50</b>	<b>3,00</b>	<b>17,00</b>	<b>925,25</b>
<b>U-TEAM</b>								<b>76,26</b>
	% 23,836427	12,025827	15,388754	42,776433	3,8203852	0,322841	1,8294323	

PRELAN - preliminary study, feasibility analysis  
 REQUIRE - requirements (functional specifications)  
 DESIGN - design (technical specifications)  
 IMPLEM - implementation, module testing  
 TEST - integration testing, system testing  
 INS,U,MAI - installation, start if use (deployment), maintenance  
 OTHER - everything else

Kuva 18: The working hours table as of 8.5.06.

Taulukko 23: Amount of code.

Programming language	Java
Total lines of code	6000
GameCore	1200
GUI	2600
Other parts (Games etc.)	2200

Taulukko 24: The documents of the project.

Document	Number of pages
Project Plan	21
Usability Plan	11
Requirements Specification	23
Implementation Plan	54
User Interface Plan	18
Test Plan	11
User Guide	6
Test Report	8
Maintenance Guide	6
Final Report	10
Final Story	5
Combined number of pages	173

## 9 MetaEdit Team

### 9.1 Overview

The purpose of the project was to develop a Fujaba plug-in application. The application exports the UML models in the XML format (GXL). The GXL format is compliant with the UML class diagram metamodel defined in MetaEdit+.

### 9.2 Organisation and management

The client of the project was Zheyang Zhang from the University of Tampere, Department of Computer Science. The project lecturer was Timo Poranen. The project did not have any project manager, but the project was managed by all the team members together. The tasks of the project manager were divided for all of the team members. The team consisted of five team members that are the following:

- Ruijie Ban, responsible for project documentation, project testing.
- Pan Pan, responsible for project documentation and project implementation.
- Zhigang Yang, responsible for project requirement, project documentation, and project implementation.
- Marko Koivu, responsible for project design, project documentation, and project implementation.
- Ilkka Vähämöttönen, responsible for project documentation, project infrastructure, and project implementation.

Ilkka maintained the project web page and collected the project hours for the hour table. Zhigang was in charge of the requirements specification. All the other tasks were done together by all the team members.

### 9.3 Methods and tools

The project had two main modelling programs that were integrated with GXL: Fujaba and MetaEdit+. Fujaba version used in the project is Fujaba Tool Suite RE edition 4.2.0 Build 20041122 with JavaParser 2.0 Build 0 installed (default). MetaEdit+ version is 4.0. MetaEdit+ version contains the UML metamodels that it supports.



Fujaba GXL plug-in is written in Java and Java Development Kit (JDK) version 1.5.0\_06 is used as the compiler. Java is a cross-platform programming language, so it runs for example in Windows XP and Linux. End users install only Java RunTime Environment (JRE) of the same Java version. Project development was made with Eclipse (IDE) version 3.1.2.

The project was done with waterfall model of software engineering projects. First the requirements were specified and a prototype was constructed. The implementation design uses object oriented approach. JUnit was planned to be used in unit testing, but the team did not consider it as a convenient way of testing the structures of the plug-in. More traditional testing was used instead like debugging, printing to console, and comparing the generated results. JUnit is recommended for testing in the long run.

## 9.4 Project phases

The project was divided into main phases according to the waterfall approach. The phases are requirements specification, design, implementation, testing, and delivery. Also a prototype was constructed and other deliverables like documentation were made. The project had documentation and project management phases that lasted for the whole duration of the project.

## 9.5 Conclusions

The project was a scientific project in nature, because there was not any actual traditional transactional system or such to be created. This project did not have lots of user interfaces or business logic. The main functionality converts UML class diagram into GXL.

It is fairly straight forward to create plug-ins for Fujaba, thanks to Fujaba plug-in architecture. The plug-in can communicate with Fujaba through Fujaba API. There was some minor version conflict with Fujaba API documentation and the downloadable Fujaba, so the inner structure of Fujaba API had to be found out explicitly or manually. It was sometimes not easy to find out what parts of Fujaba user interface mapped to what parts in Fujaba API.

Fujaba GXL plug-in was created fully according to the requirements specification. There are some features that could be added to the plug-in in the future, like the capability of exporting other UML diagram types than class diagrams to GXL.

## 9.6 Statistics

Taulukko 25: Project MetaEdit Team hour table.

	Prelana	Require	Design	Implem	Testing	Other	Total	%
Meet	40	10	10	23	8	13	104	17,8
Insp	6	8,5	11	6	12	1	44,5	7,6
Study	18	6,5	9	14,5	25,5	0	73,5	12,6
Doc	63,5	34	13,5	8,5	41	29	189,5	32,4
Pro/D	0	0	0	0	0	7	7	1,2
ProjM	0	0	0	0	0	22,5	22,5	3,9
Work	0	0	2,5	77	28	36	143,5	24,5
Total	127,5	59	46	129	114,5	108,5	584,5	
%	21,81	10,09	7,87	22,07	19,59	18,57		

Taulukko 26: Project's documents.

Document	Pages
Project plan	30
Requirements specification	18
Vision and Scope Document	5
Implementation plan	28
Test plan	25
User manual	8
Test report	4
Maintenance document	11
Final report	14
Final story	3
Total	146

Taulukko 27: Project's codelines.

Programming language	Java	%
LOC	3123	100
SLOC	1814	58.1
Comment	785	25.1
BlankLines	524	16.8
Classes	36	
TotalSize	85.00KB	

## 10 Itku

### 10.1 Yleiskuvaus ohjelmasta

Itkuhälytintä on Symbian-sovellus S80 -yhteensopiville matkapuhelimille. Itkuhälytintä on vauvanvahti, joka kuuntelee nukkuvaa vauvaa ja suorittaa hälytyksen, kun vauva herää. Se on käytettävissä heti asennuksen jälkeen, eikä vaadi monimutkaisia asennustoimenpiteitä.

### 10.2 Projektiorganisaatio

Taulukko 28: Itku-ryhmän jäsenet

Henkilö	Rooli
Esa Karvanen	Projektipäällikkö
Eero Hietaranta	Toteutussuunnitelma ja ohjelmointi
Juha Hjelm	Testaussuunnitelma ja projektisuunnitelma
Matti Pitkänen	Toteutussuunnitelma ja ohjelmointi
Samu Ristkari	Projektisuunnitelma ja ohjelmointi
Atte Sepponen	Vaatimusmäärittely ja projektin kotisivut

Projektin asiakas oli Cape Peace Software. Yhteyshenkilönä toimi Tommi Lukkarinen.

Requirements Engineering -kurssilla tehtiin harjoitustyönä Itkuhälytintäprojektille alustava vaatimusmäärittely.

Käytettävyysryhmästä saatiin apua sovelluksen käytettävyyden kehittämiseen ja käytettävyystestien tekemiseen. Itkuhälytintäprojektin yhteyshenkilö käytettävyysryhmässä oli Minna Sundström

### 10.3 Välineet, menetelmät ja tekniikat

- S80 SDK. Vaikeakäyttöiseksi osoittautunut väline, jossa oli huomattavia puutteita. Näistä erityisesti maininnan arvoinen on emulaattorin ja puhelimen ilmeinen erilaisuus ja yhteensopimattomuus.
- Visual Studio. Microsoftin ohjelmointiympäristö. Tekee tehtävänsä.
- Microsoft Office. Käyttökelpoinen työväline.

- Open Office. Tällä kirjoittelee dokumentteja, ja tekee tunti- ja aikatauluja siinä missä muillakin office-paketeilla.

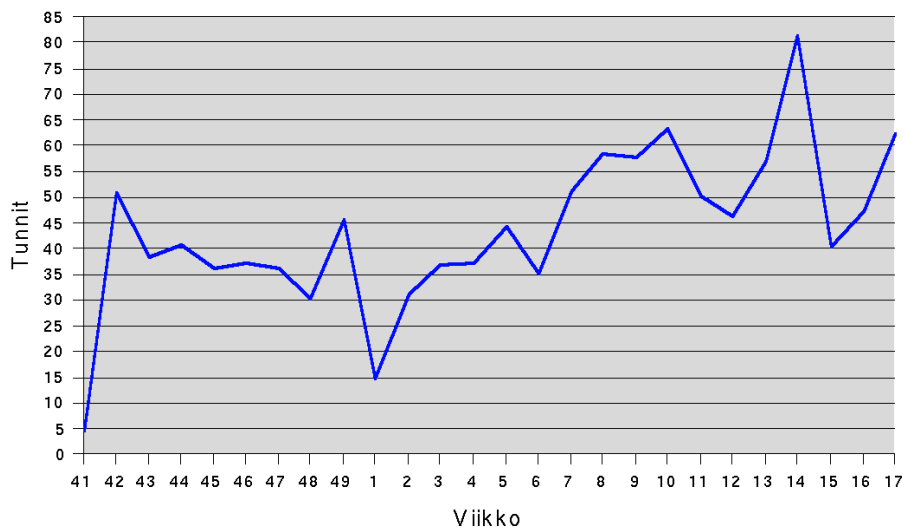
Koodia kirjoitettiin toisinaan yksilötyönä jokainen omalla ajallaan kotonaan, ja välillä laitoksen harjoittelijatilassa eräänlaista extreme programming metodologiaa noudattaen missä yksi koodasi, ja muu paikallaoleva ryhmä seurasi vieressä. Näiden ohjelmointikäytäntöjen soveltaminen osoittautui toimivaksi ratkaisuksi.

Viikottain pidettiin viikkopalaveri, jossa käytiin läpi viikon tehtävät ja annettiin uusia. Viikkopalaverissa tehtiin aina palaveriraportti, joka laitettiin projektin kotisivuille kaikkien nähtäväksi.

Palaverien lisäksi ryhmä oli yhteydessä toisiinsa sähköpostin ja kännykän välityksellä.

## 10.4 Projektin eteneminen

Projekti eteni kohtalaisen tasaisesti läpi lukuvuoden. Työmäärä kasvoi kuitenkin loppua kohden.



Kuva 19: Työn intensiteetti projektin aikana.

## 10.5 Johtopäätökset projektista

Kaiken kaikkiaan projektiryhmä sai hyvän kokonaiskäsityksen siitä millaisia vaiheita ja toimia liittyy ohjelmistoprojektiin. Hyvänä oppina jäin käteen sananlasku: ”Hyvin suunniteltu on puoliksi tehty”. Toisaalta tuo sananlasku ei pätenyt kokonaan toteutusvaiheessa. Projektin onnistumisen kannalta oli kuitenkin selvää, että huonolla suunnittelulla ei olisi päästy pitkälle.

Negatiivisessa mielessä projektin etenemiseen vaikutti todella kireä aikataulutus, jonka mukana oli vaikeuksia pysyä mukana. ”Normaalia elämää” piti muokata projektille suotuisaksi. Lisäksi etenkin Symbianin tapauksessa pelihattu meinasi revetä useaankin otteeseen.

Projekti katsotaan onnistuneeksi, koska sen puitteissa pystyttiin saamaan valmiiksi ”toimiva” S80-itkuhälytin.

## 10.6 Mietelauseita

- ”Kysy Googlelta”
- ”Projekti on täydellisessä kriisissä”
- ”Manataan Symbian-demonit pois”

## 10.7 Positiivista

Projektisuunnitelman kirjoittaminen oli erinomaisen opettavainen kokemus, vaikka sen aikataulua ei niin yllättäen pahemmin noudatettukaan. Syksyllä oli ensikertalaisen toki vaikea kuvitella, mitä kaikkea valmis sovellus lopulta sisältää ja miten paljon aikaa toteutusvaiheisiin kuluu. Vastedes osaamme suhtautua aikataulutukseen entistä kyynisemmin.

## 10.8 Negatiivista

Kurssilla kirjoitettiin aika paljon erilaisia raportteja ja dokumentteja, varsinkin kurssin alkuvaiheissa. Näissä dokumenteissa on ollut aika paljon osuuksia, jotka ovat olleet lähes muuttumattomina kaikissa dokumenteissa. Tällainen samojen asioiden toistaminen dokumentista toiseen tuntuu hieman turhalta.

Tuntitaulukoiden täyttäminen tuntui välillä hieman turhalta ja tuntien saaminen oikeaan kohtaan taulukossa tuntui välillä aikamoiselta arpomiselta. Viikkopalaveriraporttien tekeminen oli myös välillä hieman turhan tuntuis- ta, mutta nämä ovat kokonaisuuden kannalta pieniä asioita.

## 10.9 Tilastot

Taulukko 29: Itku-projektin tuntitaulukko.

	Esit	Määr	Suun	Tot	Test	As, Ko, Yp	Muu	Yht	%
Tumi	29	6,5	6	92,5	6	5,5	12	156,5	12,8
Pala	19,75	12	17	58,5	17	7,5	24	155,75	12,8
Tark	4	7,5	8	8	7		1	35,5	2,9
Opet	10	6	4,5	11	4	2	21,5	59	4,8
Doku	18	26	17,5	40	18	29	5,5	154	12,7
Pro/D								0	0
Projh	13	8	8	28,5	22	3	41,75	124,25	10,2
Työ	63,5	26	9,25	365	37,5	11	22,25	534,5	43,8
Yht	156,25	92	70,25	603,5	111,5	58	128	1219,5	
%	12,8	7,5	5,8	49,5	9,1	4,8	10,5		

Taulukko 30: Itku-projektin dokumentit.

Dokumentti	sivuja
Projektisuunnitelma	20
Projektin käytettävyyssuunnitelma	10
Vaatimusten määrittely	19
Toteutussuunnitelma	21
Käyttöliittymäsuunnitelma	10
Testaussuunnitelma	17
Käyttöohje	2
Testausraportti	7
Ylläpito-ohje	10
Loppuraportti	20
Loppukertomus	4
Yhteensä	140

Taulukko 31: Koodirivien lukumäärä.

Ohjelmointikieli	Symbian C++
LOC	2030
SLOC	2030
Uudelleenkäytetty	0
Uudelleenkäytetty ja muokattu	221
Generoitu	83
Luokkia	24

## 11 Webcom

### 11.1 Yleiskuvaus ohjelmasta

Projektin tarkoituksena luoda helposti käytettävä järjestelmä MicroMagic Oy:lle, jossa BIGblue-tietokoneita voidaan koota komponentti kerrallaan, syöttää lisää komponentteja ja selata niitä monipuolisten hakuehtojen avulla, sekä muokata käyttöoikeuksista riippuen valmiita konepaketteja ja muita eri rekistereissä olevia tietoja.

MicroMagic Oy:n asiakkaan on mahdollista ladata järjestelmästä oman koneensa komponentteihin liittyviä ajuritiedostoja ja muita koneeseen liittyviä tiedostoja. Asiakas voi rekisteröidä tietokoneensa järjestelmään, mutta tiedostojen lataaminen onnistuu myös ilman rekisteröintiä. Rekisteröinnin voi hoitaa myös koneen myyjä asiakkaan toivomuksesta esimerkiksi kaupan-tekotilanteessa. Järjestelmästä ovat nähtävissä myös koneen eri osien takuu-tiedot.

Järjestelmässä olevat osat:

- Tuoterekisteri - BIGblue-tietokoneiden komponentit
- Asiakasrekisteri - pidetään yllä BIGblue-koneensa rekisteröineiden asiakkaiden tietoja
- Henkilökuntarekisteri - ylläpidetään henkilökunnan käyttäjätietoja
- Toimittajarekisteri - komponenttien toimittajat
- BIGblue-rekisteri - valmiit tietokonepaketit

Rekistereiden käyttöä varten on rekisterien muokkaukseen soveltuva selainpohjainen käyttöliittymä.

### 11.2 Projektiorganisaatio

Esa Karvanen

- Projektipäällikkö

Jouni Laitinen

- WWW-vastaava, pitää yllä projektin verkkosivuja
- Ohjelmointi
- Toteutus suunnitelma



Juha Mattila

- Dokumenttivastaava
- Projektisuunnitelma
- Testaussuunnitelma

Sami Saarenpää

- Dokumenttivastaava
- Projektisuunnitelma
- Testaussuunnitelma

Sami Sieranoja

- Ohjelmointi
- Toteutussuunnitelma
- Käyttöliittymäsuunnittelu
- Tietoturvavastaava

Jarmo Tolvanen

- Ohjelmointi
- Toteutussuunnitelma
- Käyttöliittymäsuunnittelu
- Tietoturvavastaava

### **11.3 Välineet, menetelmät ja tekniikat**

Olellaiset välineet ja tekniikat:

- PHP 4 - Hypertext Preprocessor
- Apache 2 - web-palvelin
- MySQL 3.2 - web-palvelimen tietokanta
- XHTML 1.0 Strict - käytetty standardi sivustolle



Kuva 20: Projektin logo.

- CSS 2.1 - käytetty standardi sivustolle
- JavaScript - web-sivun toiminallisuus

Muut tekniikat ja välineet:

- phpMyAdmin - web-hallinta tietokannalle, nopeutti työskentelyä
- OpenOffice 2 - dokumenttien koostaminen, PDF-konversiot
- Internet Explorer 6 - selainohjelma
- Firefox 1.5 - selainohjelma
- Konqueror 3.5 - selainohjelma

Kukin ohjelmointiin osallistunut teki ohjelmointia itse parhaaksi näkemällään editorilla. Tiedostot tallennettiin ISO-8859-1 enkoodauksella. Tiedonsiirtoon web-palvelimelle käytettiin FTP:tä.

## 11.4 Projektin eteneminen

Projektiryhmämme piti viikkopalaverin joka viikko. Loppupuolella pidimme muutaman palaverin irkkikanavallakin ryhmäläisten kiireisyyden vuoksi. Lähes poikkeuksetta kaikki ryhmämme jäsenet olivat joka palaverissa mukana.

Valitettavasti asiakas oli mukana ainoastaan kahdessa palaverissa. Tähän asiakkaan osallistumisen vähyyteen vaikutti se, että asiakas toimii usean sadan kilometrin päässä, eikä näin ollen voinut lähteä kovin usein tapaamiseen.

Projektisuunnitelman katselmointi: 16.11.2005

Vaatimusmäärittelyn katselmointi: 15.12.2005

Testaussuunnitelman ja käyttöliittymäsuunnitelman katselmointi: 31.1.2006

Toteutussuunnitelman katselmointi: 14.2.2006

## 11.5 Projektin ongelmat

- Tuntikirjaus oli hieman hajanaista.
- RE-ryhmän mukaanympääminen toi paljon lisätöitä PP:lle.
- Vaatimusmäärittelyn teki eri ryhmä kuin oli tarkoitus.

## 11.6 Johtopäätökset projektista

Projektiryhmä toimi hyvin yhteen. Johtamisessa opittiin kokoajan lisää. Kursiin kuuluneet lukuisat dokumentit saatiin ihan hyvin kasaan. Alusta asti tiesimme, että projektissa on paljon tehtävää ja lopussahan se huomattiin, että asiat eivät aina mene niin kuin PP haluaisi. Irkkirajoitteisuuden puute todettiin positiiviseksi.

## 11.7 Tilastot

Taulukko 32: Projektin Webcom tuntitaulukko.

	Esit	Määr	Suun	Tot	Test	As, Ko, Yp	Muu	Yht	%
Tumi	7.5	8.5	20	58	12	8	42	156	14.08
Pala	37	29	21	69.5	12.5	8	5	182	16.43
Tark	5	0	3.5	2	0	0	0	10.5	0.95
Opet	2	9	7	24	4	4	4	76.5	6.9
Doku	32.5	40.5	56	22	37	29	27	244	22.02
Pro/D	10	11	0	15.5	0	0	0	36.5	3.29
Projh	14	5	6	12	8	0	0	45	4.09
Työ	8	14.5	16.5	246.5	22	4	48	357.5	32.27
Yht	116	117.5	130	449.5	95.5	51	148.5	1108	
%	10.47	10.60	11.73	40.57	8.62	4.6	13.4		

Taulukko 33: Projektin dokumentit.

Dokumentti	sivuja
Projektisuunnitelma	15
Projektin käytettävyyssuunnitelma	11
Vaatimusten määrittely	32
Toteutussuunnitelma	33
Käyttöliittymäsuunnitelma	17
Testaussuunnitelma	28
Ylläpitodokumentti	20
Loppuraportti	20
Yhteensä	176

Taulukko 34: Koodirivien lukumäärä.

Ohjelmointikieli	PHP
LOC	4600
SLOC	4000
Uudelleenkäytetty	500
Uudelleenkäytetty ja muokattu	200
Luokkia	30

## 12 Newsletter Manager Extensions

Projektin nimi: Newsletter Manager Extensions  
Toimeksiantaja: Ambientia Oy  
Projektin työnimi: NewMan

### 12.1 Projektiorganisaatio

<i>Nimi</i>	<i>Tehtävä</i>
Rami Törmä	Projektin johtaminen.
Mike Arvela	Ryhmän kotisivut, testaussuunnitelma, testiluokkien kehitys ja testien toteutus.
Aleksi Melto	Käyttöliittymäluonnokset ja ideointi, projektisuunnitelma, toteutussuunnitelma, käyttöliittymäprototyyppi, vaatimusmäärittely.
Ville Niemi	Vaatimusmäärittelyt.
Eerik Savolainen	Vaatimusmäärittely ja käyttöliittymäprototyyppi.
Aki Tuomi	Tietokantavastaava, toteutussuunnitelma, prototyypin kehitys, komponentin koodaus ja testaus.
Ivar Ekman	Käytettävyyksiin edustaja.

### 12.2 Asiakas

Ambientia Oy on sähköiseen liiketoimintaan ja -viestintään erikoistunut palveluyritys. Ambientia konsultoi, suunnittelee ja toteuttaa kansainvälisillä markkinoilla yhdessä yksityisten ja julkisen sektorin toimijoiden kanssa. Ambientian ratkaisut tukevat asiakkaiden menestystä. [[www.ambientia.net/portal/yritys/](http://www.ambientia.net/portal/yritys/)]

Asiakkaan yhteyshenkilöt:

Projektin tilaaja: Henri Sora, tuotanto- ja teknologiajohtaja

Projektin kontaktihenkilö: Matias Muhonen, sovelluskehittäjä

### 12.3 Projektin aihe

Projektin aihe oli uudistaa Newsletter Manager järjestelmää. Projekti käänsi osan PHP:llä toteutetusta vanhasta järjestelmästä J2EE-ympäristöön ja tuotteeseen luotiin uusia ominaisuuksia. Käännettäväksi ja kehitettäväksi määriteltiin komponentti, mikä toteuttaa Newsletter Managerin toiminnallisuudesta käyttäjien, tilaajien ja postituslistojen hallinnan, sekä uutiskirjeiden

lähettämisen. Käyttäjällä tarkoitetaan järjestelmän käyttäjää, mahdollista uutiskirjeen lähettäjää, ja tilaajalla uutiskirjeen tilaajaa.

Edellä mainittujen toiminnallisuuden lisäksi Newsletter Managerissa on huomattava määrä muita toiminnallisuuksia, koskien mm. uutiskirjeiden hallintaa ja lähetysten tilastointia.

Näiden toiminnallisuuden toteuttamista varten laadittiin vaatimusmäärittely. Eli projekti tuotti vaatimusmäärittelyn koko järjestelmälle ja toteutti aikaisemmin mainitun osan järjestelmästä.

Vaatimusmäärittely ja komponentin toteutuksen lisäksi laadittiin myös käyttöliittymäprototyyppi.

## 12.4 Kokemuksia projektin kulusta ja kohdatuista ongelmista

Kokonaisuutena projekti eteni ja onnistui hyvin. Tästä huolimatta projektin eteneminen ei ollut aivan kitkatonta; Projektin aikana kohtasimme erinäisiä ongelmia ja vastoinkäymisiä. Mikään vastoinkäymisistä ei kuitenkaan aiheuttanut ylitsepääsemättömiä vaikeuksia, ja osa vastoinkäymisistä voitanee laittaa normaalin projektityöskentelyn takkuilun piikkiin. Tämä ei ole selitys tapahtuneille tai puolustus vaan toteamus: Rapatessa roiskuu ja aina hajoaa jotain, kun tehdään töitä.

Pidimme viikkopalaverit TKT:n laitoksen tiloissa kerran viikossa. Viikkopalaverit olivat alussa tarpeellisia, joskin hieman tehottomia projektin punaisen langan ollessa alussa hukassa. Viikkopalaverien merkittävin funktio oli tiedon ja vastuun jakaminen ryhmän sisällä, sekä tulevan suunnittelu. Syyslukukaudella palaverit järjestettiin maanantaisin ja kevätlukukaudella perjantaisin. Uuden vuoden jälkeen huomasimme pian, että palaverit on parempi pitää maanantaina kuin perjantaina. Vähäisen osanottajamäärän takia, perjantaipalaverointi oli ajoittain hieman turhauttavaa. Projektin jäsenet eivät aina päässeet osallistumaan perjantain palaveriin mm. kotiseudulle matkustamisen vuoksi.

Viikkopalaverien lisäksi pidimme vähintään kerran kuukaudessa seuranta-palaverin toimeksiantajan kanssa. Näissä palaverissa toimeksiantajalle selvitettiin projektin sen hetkinen tila ja pyrittiin esittelemään aikaansaannoksia, kuten käyttöliittymäprototyyppi ja dokumentit tai järjestämään katselmointi.

Projektin alkupuoliskolla vallitsi pitkään tila, jossa projektiryhmä ei täysin tiennyt mitä projektilta odotettiin. Alussa projektin vaatimukset tuntuivat elävän, muuttuvan suorastaan vastakohtaisiksi. Tämä taisi johtua toimeksiantaja osapuolen eri henkilöiden Newsletter Manageria koskevien näkemys-

ten yhteensulautumisprosessista. Projektiryhmän esittämät ideat ja kritiikki toivat lisää säpinää tähän sulautumisprosessiin. Suunnan löytymistä olisi voinut helpottaa projektiryhmän avoimempi kysely ja sen myöntäminen, että kaikkea mitä vaaditaan ei ymmärretty täysin.

Dokumentteja kirjoitettiin ahkerasti koko projektin keston ajan, yhteensä 244 sivua + pöytäkirjat. Kattava dokumentaatio osoittautui erittäin hyödylliseksi, suurehkojen kokonaisuuksien hahmottamisen ja valitun linjan pitämisen kannalta.

Käyttöliittymäprototyyppiä varten kehitettiin iteratiivisesti paperiprototyyppi, jonka viimeinen versio arvioitiin heuristisesti. Paperiproton ja arvioinnin pohjalta toteutettiin XHTML, CSS 2 ja JavaScript tekniikoilla selattava, ei toiminnallinen, käyttöliittymäprototyyppi.

Toteutus- ja testausvaihe sujui suuremmista ongelmista, sen jälkeen kun tuotanto saatiin käyntiin. Toteutusympäristön toimituksen viivästymisen takia, ohjelmointia ei voitu aloittaa aikataulun mukaan. Toteutusympäristön viivästymisen johtui laitteistotoimittajan ongelmista. Ohjelmointi- ja testausvaihe oli opettavaista sen suorittajille heille uusien työkalujen ja tuotantomenetelmien ansiosta. Uusien tekniikoiden opettelu koettiin osittain vaivaloiseksi, mutta samalla opettavaksi ja motivoivaksi. Uutta opittiin mm. yksikkötestauskokeista, ohjelmointityön käytännön järjestelyistä ja työkaluista.

## 12.5 Käytetyt työvälineet, työmenetelmät

Seuraavassa listassa esitetään projektissa käytetyt ohjelmistot, työkalut ja tekniikat.

- Java 1.5 JRE
- MySQL 5
- Tomcat 5.5
- Postfix 2.2.4
- Apache 2.0.54
- Mod\_jk
- Hibernate 3.0.5
- Spring 1.2.6
- JSTL 1.1

- MySQL-Connector / J 3.1.12
- FieldCollection
- Eclipse
- Subversion
- Junit 3.81
- Cactus
- Javadoc 1.5
- XHTML 1.0
- CSS 2.0
- JavaScript

## 12.6 Tilastot

Työtunnit janaesityksenä:

SYKSY 2005

VKO 41, 42, 43, 44, 45, 46, 47, 48, 49, 50

TUNNIT 14, 56, 53, 37, 58, 41, 27, 77, 90, 27

Kevät 2006

VKO 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17

TUNNIT 9, 22, 29, 44, 63, 70, 51, 48, 30, 15, 1, 18, 49, 32, 41, 35, 28

	Esitutkimus	Määrittely	Suunnittelu	Toteutus	Testaus	AS, KO, YP	Muu	Yht. h.	Yht. %
DOKU	2,5	87	54,75	112,5	6,75	-	2	265,5	24,9%
OPET	16,25	8	3	13,75	15,5	-	4	60,5	5,7%
PALA	46	73	47,5	83,9	23	-	5	278,4	26,2%
PROD	-	-	80,5	12	-	-	-	92,5	8,7%
PROJH	2,5	16,75	13,25	22,25	7	-	1	62,75	5,9%
TARK	2	14	11	9	3,5	-	-	39,5	3,7%
TUMI	8,25	17,5	17,5	27	13	-	-	83,25	7,8%
TYÖ	6	48,5	62	47	14,5	-	3	181	17%
Yhteensä / h.	83,5	264,75	289,5	327,4	83,25	-	15	1063,4	-
Yhteensä / %	7,8%	24,9%	27,2%	30,8%	7,7%	-	1,4%	100 %	100%

Taulukko 35: Työlajit ja elinkaarivaiheet prosenttiosuuksina kokonaistyötuntimäärästä.



<i>Dokumentti</i>	<i>sivumäärä</i>
Projektisuunnitelma	61
Vaatimusmäärittely	63
Käyttöliittymädokumentti	18
Käytettävyysuunnitelma	20
Testaussuunnitelma	23
Toteutussuunnitelma	29
Loppuraportti	30

Koodia: n. 3000 riviä  
Kommentteja: n. 900 riviä

## 13 SMS/E-mail yhdyskäytävä

### 13.1 Yleiskuvaus ohjelmasta

Yhdyskäytävä mahdollistaa sekä tekstiviestien lähetyksen ja vastaanoton GSM-puhelimilla, että sähköpostiviestien haun, varastoinnin ja välityksen. Viestien välitykseen tarjotaan SOAP-standardia käyttävä rajapinta, joten palvelun graafiset käyttöliittymät voidaan toteuttaa millä tahansa standardia tukevalla sovelluskehitystyökalulla.

### 13.2 Projektiorganisaatio

Similä Jukka

Käyttöjärjestelmien ja ohjelmistojen asiantuntija. Asensi projektin käyttö- ja kehitysympäristön. Toteutti tietokantarajapintaliittymän ohjelmoijille. Opasti muita ohjelmoijia projektin aikana ja korjasi osan bugeista.

Hannu Lohtander

Ohjelmistoasiantuntija. Toteutti sähköpostiviestien lähetyksen ja vastaanotto-osion. Korjasi SOAP-rajapintafunktioita.

Kristian Mattila

Suunnitteli ja teki projektin kotisivut. Toteutti SOAP-rajapinnan admin-funktiot ja käyttöliittymän yhdyskäytävän konfiguroimiseen.

Henri Aalto

SOAP-rajapinnan käyttämien käyttöliittymien suunnittelu ja toteutus testikäyttöön, sekä esittelyyn. Rajapinnan dokumentointi ja suunnittelu ohjelmointitiimin kanssa.

Petri Vuorinen

Projektipäällikkö. SOAP-rajapinnan sms- ja email-toimintojen ohjelmointia.

### 13.3 Välineet, menetelmät ja tekniikat

#### Linux

Linux valittiin yhdyskäytävän käyttöjärjestelmäksi vapaan käytön ja muokattavuuden takia. Myös tekstiviestien välitysohjelmia löytyi useampi vaihtoehto Linuxille kuin Windowsille.

#### PHP

Yhdyskäytävän ohjelmien toteutuskieleksi suunniteltiin ensin Javaa. Koska käyttöjärjestelmään oli joka tapauksessa kirjoitettava tausta-ajossa olevia skriptejä, päätettiin ohjelmointikieleksi ottaa PHP, jotta toiminnallisuuksien käyttö olisi eri osien välillä mahdollisimman helppoa.

#### SMS Server Tools

SMS Server Tools valittiin tekstiviestien palvelinohjelmistoksi, koska toimikiantaja halusi käyttää viestien välitykseen GSM-puhelimia. Lisäksi SMS Server Tools:n avulla voitiin viestien lähetys toteuttaa erilliseksi osa-alueekseen, joka voidaan haluttaessa korvata suoralla yhteydellä puhelinoperaattorin tekstiviesti-välityspalvelimeen.

#### PostgreSQL

PostgreSQL-ohjelmistoa käytettiin toteutusvaiheessa simuloimaan tuotantokäytön tietokantaa.

#### SharpDevelop

Käyttöliittymiä suunniteltiin aluksia Visual FoxPro:lla toteutettaviksi, mutta SOAP-rajapinnan käyttö FoxPro:lla osoittautui vaikeaksi. Joten käyttöliittymät päätettiin toteuttaa .NET:llä ja PHP:lla. SharpDevelop-ohjelmaa käytettiin .NET-ohjelmointiympäristönä sen avoimuuden ja helppokäyttöisyyden vuoksi.

## 13.4 Projektin eteneminen

Projekti alkoi lokakuun 2005 lopulla toimeksiantoon tutustumisella. Requirements Engineering -kurssin opiskelijat olivat valmistelleet alustavan vaatimusmäärittelyn, jota projektiryhmä joutui vielä tarkentamaan ja korjaamaan todella paljon. Vaatimusmäärittelyn lukkoon lyöminen kesti liian pitkään, ja ryhmän olisi kannattanut neuvotella vaatimuksista enemmän asiakkaan kanssa. Vaatimusmäärittelyyn tehtiin muutoksia vielä ryhmän sisäisesti, kun huomattiin että joitain asioita ei oltu huomioitu lainkaan. Vaatimusmäärittely katselmoitiin joulukuussa.

Projektisuunnitelma katselmoitiin marraskuun lopulla. Projektin edistyessä suunnitelmaa ei noudatettu kovinkaan tiukasti, eikä sitä päivitetty riittävän ahkerasti. Joulun ja vuodenvaihteen aikoihin projektin etenemisessä oli noin kolmen viikon tauko.

Toteutusvaihe alkoi virallisesti helmikuun lopulla, mutta erilaisia toteutukseen liittyviä kokeiluita oltiin tehty jo usean viikon ajan. Testaussuunnitelma katselmoitiin helmikuussa ja toteutussuunnitelma maalikuussa.

Projektin alkupuolella suurin osa dokumenteista tehtiin huolellisesti ja saatiin valmiiksi annettujen aikarajojen puitteissa. Toteutusvaiheen alettua dokumentointiin käytettiin vähemmän resursseja, koska toteutuksen edistymisen kannalta kaikki aika oli ohjattava itse työn tekemiseen. Tuotteen valmistuminen viivästyi alkuperäisestä aikataulusta merkittävästi, ja kaikkia vaatimusmäärittelyn kohtia ei ehditty toteuttamaan. Olennaisimmat osat kuitenkin valmistuivat toukokuun 2006 loppuun mennessä. Testausvaiheen varsinainen läpikäyminen jäi myös varsin vähälle huomiolle, mutta suurin osa komponentti- ja integraatiotestauksesta tuli tehtyä toteutuksen lomassa.

Toteutusvaiheen loppupuoliskolla tultiin siihen tulokseen, että käyttöliittymien toteutukseen alunperin suunniteltua työkalua (Microsoft Visual FoxPro) ei saada toimimaan yhdyskäytävän kanssa täydellisesti. FoxPro-ongelmat johtuivat Microsoft SOAP Toolkit -kirjaston olioiden käsittelyyn liittyvistä ongelmista. Näin ollen käyttöliittymien toteutusvälineeksi valittiin SharpDevelop-niminen avoimen lähdekoodin ohjelma ja mallikäyttöliittymät toteutettiin C#.NET-kielellä.

Toteutusvaihe venyi lopulta kurssilla asetettujen aikarajojen yli. Toukokuun puolessavälissä todettiin yhdessä, että projekti ajetaan alas hallitusti, koska aika loppui kesken. Toteutettavia ominaisuuksia karsittiin, mutta jo toteutetut toiminnot viimeisteltiin siten, että ohjelmistoa pystytään käyttämään.

### 13.5 Johtopäätökset projektista

Projektin etenemistä on haitannut yletön kirjoitettavien dokumenttien määrä. Ajanhallintaan keskityttiin projektin alkuvaiheessa paremmin kuin keväällä, mutta tämä johtui mahdollisesti dokumenttien palautusten takarajoista ja näin ollen pikemminkin haittasi kuin auttoi projektin työn toteutusta. Kun työtä päästiin koodaamaan, päälle puski testaussuunnitelman kirjoittamista ynnä muuta, ja varsinaiselle ohjelmoinnille on jäänyt riittämätön määrä tunteja tämänlaajuisen projektin kunnolliseen toteuttamiseen.

Ohjelmoinnissa törmättiin myös sellaiseen ongelmaan, ettei millään kurssilla ole saatu valmiuksia usean ohjelmoijan samanaikaiseen työskentelyyn samojen tiedostojen kanssa. Versionhallintaa on luonnollisesti käytetty ohjelmoinnin apuna, mutta monelle projektimme jäsenistä tämä kurssi on ensimmäinen jolla minkäänlaiseen versionhallintaohjelmistoon on törmännyt. Tästä on aiheutunut joidenkin ohjelmoiden toimetttömyyttä, kun on pitänyt odotella toisten toteuttavan osia funktioista.

Jos projekti aloitettaisiin nyt, tulisi ajankäyttö suunniteltua varmasti huolellisemmin. Saattaisimme myös karsia asiakkaan toiminnallisista vaatimuksista heti alkuvaiheessa osan pois, sillä esimerkiksi koko sähköpostitoiminnallisuuden poistaminen projektista olisi tehnyt toteutetusta ohjelmistosta paremmin hallittavan ja ylläpidettävän ja jättänyt aikaa kunnolliselle ja riittävälle testaamiselle.

## 13.6 Tilastot

Taulukko 36: Projektin tuntitaulukko.

	Esit	Määr	Suun	Tot	Test	As, Ko, Yp	Muu	Yht	%
TUMI	21.5	17	26.5	34.5	0	0	6	105.5	11.43
PALA	40.2	54	122.5	79	8.8	0	0	304.5	32.98
TARK	7	5.5	1	2	1	0	0.5	17	1.84
OPET	25.5	1	15	5	3	0	14.5	64	6.93
DOKU	21	27.5	39	9.5	0	0	0	97	10.51
PRO/D	0	0.5	0	0	0	0	0	0.5	0.05
PROJH	0	0	0	0	0	0	0	0	0.00
TYÖ	3	0.5	29.5	264.8	6.5	0	30.5	334.8	36.26
YHT	118.2	106	233.5	394.8	19.3	0	51.5	923.3	
%	12.80	11.48	25.29	42.76	2.09	0.00	5.58		

Taulukko 37: Koodirivien lukumäärä.

Ohjelmointikieli	PHP
LOC	11300
SLOC	7800
Uudelleenkäytetty	3100
Uudelleenkäytetty ja muokattu	20-50
Luokkia	16

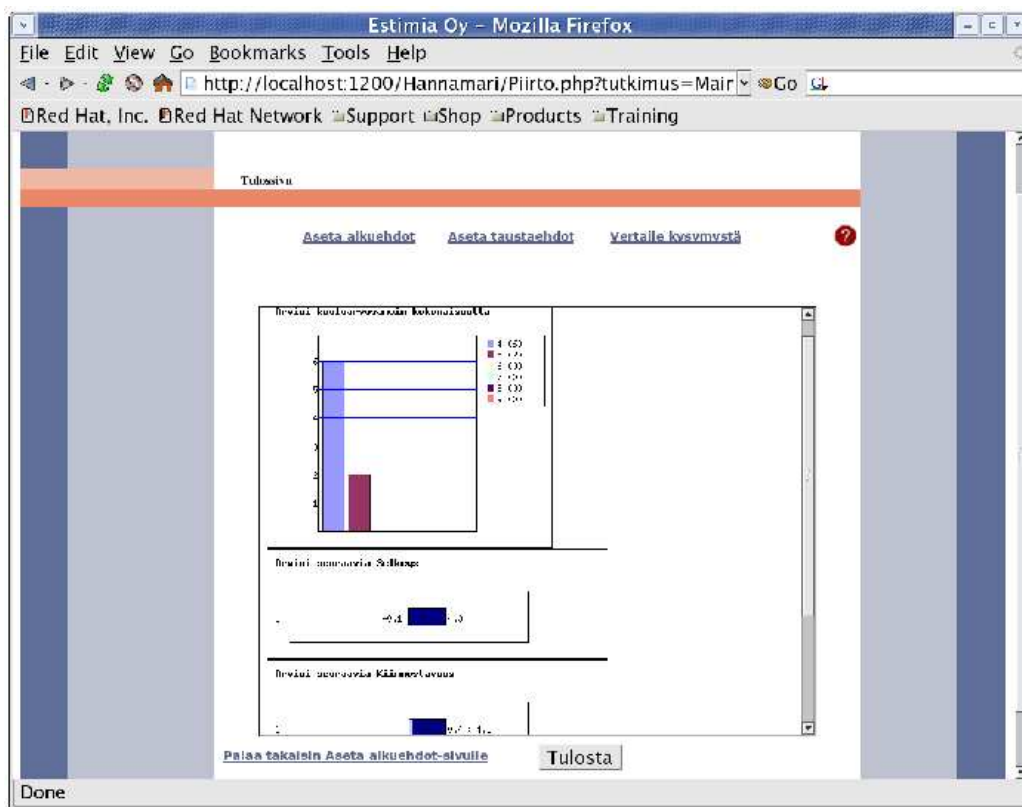
Taulukko 38: Projektin dokumentit.

Dokumentti	sivuja
Projektisuunnitelma	44
Projektin käytettävyyssuunnitelma	0
Vaatimusten määrittely	31
Toteutussuunnitelma	45
Käyttöliittymäsuunnitelma	0
Testaussuunnitelma	20
Käyttöohje	NA
Testausraportti	NA
Ylläpito-ohje	NA
Loppuraportti	NA
Loppukertomus	6
Yhteensä	146

## 14 Data visualisation

### 14.1 Yleiskuvaus IntelliToolGraph tuotteesta

Projektin päämääränä oli suunnitella ja toteuttaa jo olemassa olevaan Estimia IntelliTool -järjestelmään integroitava sovellus. IntelliTool on selainpohjainen sovellus internetin kautta tapahtuvan tutkimustiedon keruuseen, analysointiin ja raportointiin. Myös projektissa tehdyn tuotteen käyttöliittymä on selainpohjainen. Tuotteen tarkoituksena on helpottaa tutkimusten raportointia. Sovellus muodostaa yksinkertaisia kuvaajia tutkimuksista saadusta vastausdatasta siten, että kunkin kysymyksen vastausjoukko on omana kokonaisuutenaan. Kuvaajan tyyppiä käyttäjä saa valita joko pylväs- tai sektori-diagrammin. Kuiluanalyysi piirretään automaattisesti aina, kun tutkimuskysymys oli sen luonteinen, että kuiluanalyysi voidaan piirtää.



Kuva 21: Esimerkki kuvaajan piirtämisestä.



Taulukko 39: Projektiryhmän jäsenet ja heidän vastuualueensa.

Projektiryhmäläiset	Asema	Vastuualue
Marko Elo	Projektipäällikkö	Projektin johtaminen.
Joakim Hilska	Ryhmän jäsen	www-sivujen ylläpito ja ohjelmointi.
Mikko Kuru	Ryhmän jäsen	Suunnittelu ja tietokantaohjelmointi.
Tuuli Laivo	Ryhmän jäsen	Projektin ohjeistus ja ohjelmointi.
Pauliina Paarlahti	Ryhmän jäsen	Tilastoanalyysien tarkistus ja ohjelmointi.
Hannamari Saarenpää	Ryhmän jäsen	Laaduntarkkailu ja ohjelmointi.

## 14.2 Organisointi ja hallinta

Asiakkaana toimi Estimia Oy. Yheyshenkilöinä toimivat Sami Peltonen, Aki Salminen ja Yrjöjuhana Heikkilä.

Käytettävyyssiimistä vastuuhenkilönä oli Jaana Huotari. Käytettävyyssiimi teki käytettävyyssuunnitelman ja käytettävyystestit.

Projektipalavereja pidettiin syksyllä viikoittain, aina keskiviikkoisin alkaen klo 16.00. Keväällä ajankohta oli sama, mutta tapaamistiheys muutettiin joka toiseen viikkoon. Palaverien harventamisella pyrittiin saamaan lisää aikaa toteutukseen ja muuhun tuottavaan työhön.

Syksyn tapaamisissa asiakas oli kaikissa mukana, kevään tapaamisissa kolme kertaa. Myös iso osa kevään tapaamisista peruttiin sairastumisen takia.

Viikoittain tehtiin viikkoraportti, mikä lähetettiin kurssin vetäjälle. Näissä projektipäällikkö mainitsi omia tuntemuksiaan projektin tilasta. Viikkoraportteja ei lähetetty kenellekään muulle.

Kuukausittain lähetettiin myös kuukausiraportti, joissa aiheina oli vaihtelevia projektityöhön liittyviä kysymyksiä. Kysymykset olivat kurssin vetäjän tekemiä.

Projektin dokumenteista tehtiin viiden dokumentin kurssin vaatimat pakolliset katselmoinnit. Jokaiselle katselmoitavalle dokumentille valittiin projektiryhmän jäsen dokumentin tuottajaksi.

Taulukko 40: Projektin eteneminen vaiheittain.

Aika	Vaihe
8.10.2005 - 16.11.2005	Esitutkimus
17.11.2005 - 13.12.2005	Määrittely
14.12.2005 - 15.2.2006	Suunnittelu
16.2.2006 - 26.4.2006	Toteutus
27.4.2006 - 16.5.2006	Testaus ja korjaus

### 14.3 Työkalut ja menetelmät

Melkein kaikki työkalut olivat lähes alusta asti selvillä. Ainoastaan kehitysympäristön kanssa oli ongelmia, koska sen haluttiin olevan helposti kaikkien ryhmäläisten käytössä ja vastaavan mahdollisimman paljon asiakkaan ympäristöä.

Alla on lista merkittävimmistä käytetyistä työkaluista.

- Office 2003 tuoteperhe. Käytettiin dokumenttien laatimiseen. Toimi hyvin. Projektipäällikkö käytti myös openoffice 2.0:aa.
- CUTEpdf Käytettiin dokumenttien muuntamiseen pdf-tiedostoiksi. Erittäin toimiva.
- Kehitysympäristö Lopullinen kehitysympäristö saatiin Tampereen yliopiston virtuaalipalvelimelta (vista.cs.uta.fi). Pieniä ongelmia ilmeni yhteyden saamisessa palvelimelle (myös asiakkaalla), mutta pääasiassa toiminut erittäin hyvin kehitysympäristönä projektin tarpeisiin.
- Selaimet Internet Explorer 5.0 ja Mozilla Firefox 1.0 - selaimet.

Projekti vietiin läpi muunneltua vesiputousmallia käyttäen (päätetty jo Projektisuunnitelmassa). Vesiputousmalli oli käytössä, koska se oli kurssin vaatimus. Poikkeuksena oli, että ohjelmointi aloitettiin ennen vaatimusmäärittelyn ja toteutussuunnitelman valmistumista, koska käytettävissä oleva aika on niin rajallinen.

### 14.4 Projektin eteneminen

Projektin suunnittelu- ja toteutusvaihe kestivät kauimmin. Asennusvaiheeseen ei kunnolla ehditty. Ylläpito- ja asennusohjeet saatiin kuitenkin valmiiksi.

## 14.5 Päätelmät

Projektin alussa esiintyy yleensä väärinkäsityksiä, ja asioita kannattaa silloin varmistella useampaan kertaan. Lisäksi kurssin alussa on luettavana suuri tietomäärä, jota ei kaikkea pysty muistamaan. Siksi asioita kannattaa tarkastella useasti, ettei mitään olennaista lipsahda ohi. Tämä kostautuu jälkikäteen, kuten raportoinnissa kävi. Raportointi tuli tehtyä, mutta siihen tuli käytettyä liikaa aikaa.

Toteutussuunnitelman tekemiseen voisi käyttää apuna ulkopuolista tahoa. Jos toteutussuunnitelman antamat suuntaviivat ohjaavat virheelliseen kehityssuuntaan, siitä voi lopussa joutua maksamaan kovan hinnan. Samoin myös yhteydenpito on erittäin tärkeää, ja tulisi tapahtua joka tahoa kohtaan viikoittain.

## 14.6 Tilastot

Taulukko 41: Työtuntikirjanpilotaulukko Visualisation for statistical data projektille.

	PREL	REQ	DES	IMPL	TEST	INS,U,MAI	OTHER	Total	%
PLAN	13	10	10	28	7	5	0	73	6,9
MEET	35	43	18	28	3	2	6	135	12,8
INSP	8	15	19	5	8	0	5	60	5,7
STUDY	5	3	5	6	3	3	9	34	3,2
DOCUM	22	39	62	29	61	32	18	263	24,9
PROD/D	16	15	9	0	0	0	0	40	3,8
PROJM	16	7	18	24	1	5	2	73	6,9
WORKJ	35	16	11	164	42	24	85	377	35,7
Yhteensä	150	148	152	284	125	71	125	1055	
%	14.2	14.0	14.4	26.9	11.8	6.7	11.8	100.000	

Taulukko 42: Projektin koodirivien määrä.

Ohjelmointikieli	PHP
LOC	3750
SLOC	3456

Taulukko 43: Syksyn Viikottaiset työtunnit.

Viikko	41	42	43	44	45	46	47	48	49	50	51	52
Tunnit	7	27	32	34	57	45	26	32	26	47	3	3

Taulukko 44: Kevään Viikottaiset työtunnit.

Viikko	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
Tunnit	18	19	45	37	35	23	44	20	17	21	30	44	16	22	25	35	30	48	44	10	6	37

Taulukko 45: Projektissa tuotetut dokumentit.

Documentit	Sivua
Projektisuunnitelma	39
Käytettävyysuunnitelma	18
Vaatimusmäärittely	19
Toteutussuunnitelma	16
Kilpailija analyysi	9
Käyttöliittymäsuunnitelma	18
Testaussuunnitelma	41
Käyttöohje	16
Käytettävyystestiraportti	4
Ylläpito ohje	13
Testausraportti	33
Loppuraportti	28
Loppukertomus	4
Yhteensä	236

## 15 MUPEMAP

### 15.1 Overview

MUPEMAP is an Open Source world map database for the Multi-User Publishing Environment (MUPE) created by Nokia Research Centre. MUPE platform is an Open Source application platform for creating mobile multi-user context-aware applications. MUPE is a client-server system where the clients are run in mobile devices and the server is running in the Internet. The MUPEMAP database platform was created in standard MUPE application fashion using MUPE XML server-side scripting and Java. The server uses an SQL database for storing the world map and game data respectively. The application UI operates on MIDP 2.0 enabled mobile devices running the MUPEClient.

The game application is a location-based application which uses the MUPEMAP database and the MUPE platform.

### 15.2 Organisation and management

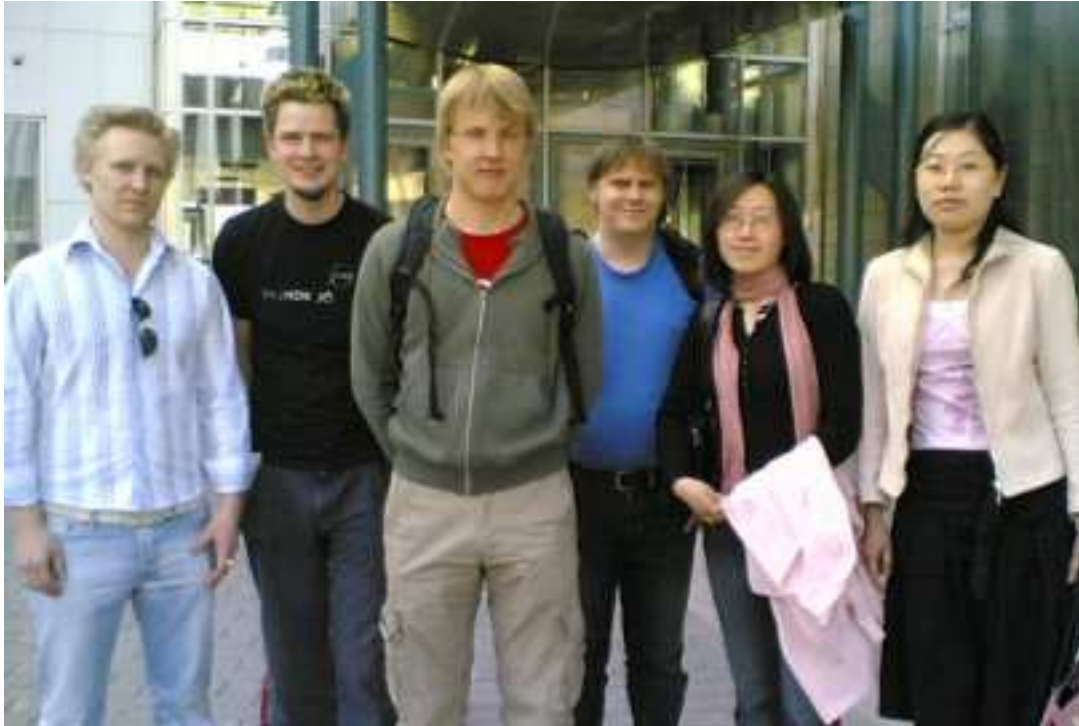
The MUPEMAP team consists of the following members and speciality areas:

- Teemu Mäki, team lead
- Jouni Erola, positioning and server programming
- Hannes Karhumaa, client programming
- Sakari Tamminen, databases
- Xiaoqing Meng-Pitkänen, testing
- Wei-Wei Zhang, UI and graphics design

The project group had semi-regular meetings on almost once a week. Meetings with the client were organised when necessary. The project members reported their hours to the project manager using a semi-automatic web-based interface.

### 15.3 Methods and tools

- Fujaba+DBSchema - a free solution which was used to create parts of the database code and SQL-files automatically. However, it contained some errors and it was necessary to alter the source code of the tool to make it work properly.



Kuva 22: Project members.

- Eclipse and MySQL Query Browser - worked OK for their purpose
- XMLBuddy Eclipse plugin - not much use as there was not much static XML editing required (or possible)
- PBWiki - used as an online collaboration tool and file storage system. This solution was quite satisfactory, although some aspects of the wiki - most notably the table syntax - caused some difficulties.
- A web-based tool for reporting hours to the project manager. The project manager extracted the data manually, but the tool ensured all reports could easily be found in one place.

## 15.4 Project phases

The beginning stages of the project went more or less as planned. The beginning of the project was delayed a bit by the absence of the client contact, but this delay was easily caught up. The requirements engineering course group produces some input for the requirements elicitation stage, however

the course schedules were not optimal and quite a lot of work was needed to finalise the requirements specification properly. A working UI prototype was in place for the autumn term presentation.

Implementation stage started a bit later than was planned due to difficulties with schedules in the beginning of 2006, but it was kicked off successfully with a game design workshop in Nokia's premises.

The schedule slippage started upon entering the coding stage. The MUPEMAP platform was not as simple as anticipated, and some database driver related problems arose. Also, Nokia's own sales start was badly postponed for the N80 which was supposed to be the reference hardware for the MUPEMAP database. The phones were never obtained by the MUPEMAP team. Since there was no proper code to test, the testing stage was pushed to the very end of the project (beginning of May). Due to problems with the database code, the project scope was redefined not to include the sample application, which was left unfinished by the end of the project work course.

## 15.5 Conclusions

The co-operation with the client went went up to the very end of the project. The support from the client was quite adequate at first - there was an organised game design workshop. Secondary hardware, although without positioning support, was provided to the group. The problems with the client support began when the N80 sales start was postponed. Also, the client support was worst at the most critical - coding - stage. This was due to the main contact at Nokia going on paternal leave earlier than was expected and the backing up personnel was not able to give adequate support when it was most needed.

The project group felt there was too much paperwork involved as opposed to "real" work but this was understood as being part of the exercise. The co-operation between the group was not optimal when compared to a real software company as there were no common working facilities, no common collaboration tools (except for the wiki) and not always common schedules either. In conclusion, most problems faced were such that would probably not arise in a real software company working environment. It was felt that the internal communications were usually too slow and difficult.

The group agreed that a partially failed project serves better as a learning experience than a project which faces no problems at all.

## 15.6 Statistics

Taulukko 46: Working hour table of project MUPEMAP.

	PREL	REQ	DES	IMPL	TEST	Total	%
Planning	2	4	10	22	0	38	5,6 %
Meetings	48	24.5	44	52.5	0	179	26,3 %
Inspections	0	4	12	0	0	16	2,3 %
Studying	27	41.5	18.5	34	0	120	17,6 %
Documenting	2	15.5	53.5	52	0	125.5	18,4 %
Prototype / demo	0	0	7	2	0	9	1,3 %
Project mgmt	10	13	15	26	0	64	9,4 %
Work	0	0	20.5	109	0	129.5	19,0 %
<b>TOTAL</b>	<b>89</b>	<b>77.5</b>	<b>180.5</b>	<b>335.5</b>	<b>0</b>	<b>681</b>	<b>100,0</b>

Taulukko 47: Project's documents.

Document	Pages
Project plan	Wiki format, ca. 16 printed pages
Project's usability plan	14 pages
Requirements specification	15 pages
Implementation plan	41 pages
Test plan	Wiki format, ca. 17 printed pages
Test report	11 pages
Maintenance document	Wiki format, ca. 14 pages
Final report	Wiki format, ca. 6 pages
Final story	4 pages
<b>Total</b>	<b>ca. 138 pages</b>



Taulukko 48: Project's codelines.

Ohjelmointikieli	Java / MUPE XML
LOC	3046 LOC in package org.mupemap.db 2346 LOC in com.nokia.mupe.server 1719 LOC of static XML
SLOC	3046 SLOC in package org.mupemap.db 1136 SLOC in com.nokia.mupe.server
Reused code	N/A
Reused and modified code	
N/A	
Classes	45 classes in org.mupemap.db 23 in com.nokia.mupe.server

## 16 Korttipaikka

### 16.1 Yleiskuvaus projektista

Korttipaikka-projektin tarkoituksena oli luoda Tampereen Ammattikuvalle käyntikorttien luonti- ja tilausjärjestelmä. Tampereen Ammattikuva on Tampereella toimiva digipaino.

Lähtötilanteessa Tampereen Ammattikuvalla oli verkossa Korttipaikkaniminen käyntikorttien suunnittelu-, hallinnointi- ja tilausjärjestelmä, joka oli suunnattu ainoastaan suurehkoille asiakkaille, lähinnä yrityksille ja julkisen sektorin organisaatioille. Pienten asiakkaiden, kuten yksityishenkilön tai itsenäisen yrittäjän, käyntikorttien suunnittelu ja tilaus toteutettiin suoraan sähköpostin välityksellä, yrityksen verkkosivuilta löytyvän tarjouspyyntöpalvelun avulla tai käymällä liikkeessä. Pienasiakkaan oli neuvoteltava Tampereen Ammattikuvan työntekijän kanssa käyntikorttinsa ulkoasusta ja mahdollisesti toimittaa käyntikorttiin tulevaa omaa kuvamateriaalia. Tämän jälkeen määriteltiin halutulle korttimäärälle hinta, toimitusaika, maksutapa sekä aika. Entinen pienasiakasjärjestelmän prosessi toteutettiin siis kokonaan ihmisvoimin.

Olemassa olevan Korttipaikan rinnalle rakennettiin uusi korttipaikkatuote, joka on suunnattu pieniä määriä satunnaisesti tilaaville asiakkaille. Ammattikuvan näkökulmasta sovelluksen käyttö on automatisoitu siten, että valmiit ja maksetut sähköpostilla välitetyt tilauspyynnöt odottavat korttien painatusta. Projektin tarkoituksena oli luoda selkeä ja käytettävyydeltään helppo järjestelmä, jolla asiakkaat voivat vaivattomasti tilata ja muokata käyntikortteja.

Kyseessä on www-selaimella käytettävä käyntikorttien tilausjärjestelmä, jonka kielenä on suomi. Palvelussa asiakas valitsee haluamansa käyntikorttipohjan, johon hän syöttää haluamansa yhteystiedot. Asiakkaalla on mahdollisuus valita kortin fontti. Tämän lisäksi asiakas pystyy muokkaamaan oman nimensä, sekä yrityksen nimen kirjasinkokoa. Korttiin on myös mahdollista lisätä kuva. Tämän jälkeen asiakas lisää toimitustiedot ja tilattavan korttimäärän. Kortin muotoilu ja tilausvahvistussivuilla asiakkaalla on mahdollisuus esikatsella kortin lopullinen versio. Lopuksi asiakas maksaa tilaamansa kortit Nordean, Osuuspankin tai Sampon verkkomaksujärjestelmien kautta. Tilausprosessi päättyy sivulle, jolla kiitetään tilauksesta.

### 16.2 Projektioorganisaatio

Projektipäällikkö

- Pasi Pekkanen

## Projektiryhmän muut jäsenet

- Juha Heinonen
- Tanja Malmberg
- Pekka Peltola
- Vilppu Tuominen
- Laura Vuorenoja

Projektin päällikkönä toimi Pasi Pekkanen. Päävastuu ohjelmoimisesta oli Vilppu Tuomisella, käyttöliittymän suunnittelusta ja toteutuksesta Tanja Malmbergilla ja Laura Vuorenojalla sekä testauksesta Pekka Peltolalla. Kaikki projektin jäsenet osallistuivat kuitenkin kaikkiin projektin eri vaiheisiin aktiivisesti.

Projektin käynnistyi lokakuussa 2005. Siitä lähtien projektiryhmä on viikottain tavannut projektipalaverissa. Viikkopalavereissa on pidetty pöytäkirjaa, jotka palaverin jälkeen ovat olleet löydettävissä projektin kotisivuilla. Myös viikkoraportit, joihin on kirjattu projektin edistyminen, ovat olleet projektin kotisivuilla. Muutoin projektiryhmä on pitänyt yhteyttä lähinnä sähköpostin välityksellä.

### **16.3 Välineet, menetelmät ja tekniikat**

Projektin pääasiallisena ohjelmointivälineenä toimi tekstieditori, jonka avulla järjestelmässä käytettävät HTML-sivut, PHP-funktiot ja -skriptit sekä korttien luomiseen tarvittavat XML-määrittelytiedostot toteutettiin. Ympäristön WWW-palvelimena toimi Apache, jonka FOP-moduulia käytettiin käyntikortin esikatselussa ja painamisessa tarvittavien PDF-tiedostojen generoimiseen.

### **16.4 Projektin eteneminen**

Projektin käynnistyi lokakuussa 2005. Projektiryhmän tavattua asiakas ensimmäisen kerran nopeasti projektin käynnistyttyä, projektin suunnittelu pääsi alkamaan. Projektisuunnitelmaa tehdessä ryhmä oli vielä vähän hämillään siitä, mitä tässä oikein ollaan tekemässä, mutta vaatimusmäärittelyvaiheessa työssä päästiin kunnolla vauhtiin. Koko ryhmä oli ohjelman toimintojen suunnittelussa mukana, ja joulukuun mennessä ryhmälle olikin muodostunut jo selkeä käsitys siitä, mitä ollaan tekemässä.

Joulun jälkeen alettiin suunnitella itse toteutusta ja testausta. Myös käyttöliittymää alettiin toteuttaa HTML-kielellä vaatimusmäärittelyn näytönkuvien pohjalta. Toteutus- ja testaussuunnitelmien valmistumisen jälkeen myös ohjelman käyttöliittymä oli hyvällä mallilla, joten itse toiminnallisuuden toteuttaminen toteutussuunnitelman pohjalta pääsi alkamaan aikataulussa. Toteutus oli valmiina suunnitelmien mukaisesti huhtikuun aikana, ja testauksessa ilmenneet virheetkin ehdittiin korjaamaan. Toukokuuhun mennessä ohjelma toimi suunnitelmien mukaisesti, ja ryhmä oli tyytyväinen tuotukseensa.

## 16.5 Johtopäätökset projektista

Projektiryhmä oli kaiken kaikkiaan tyytyväinen projektin kulkuun. Projektiryhmän hyvään henkeen ja tulokseen vaikutti kaikkien positiivinen ja työtäpelkäämätön asenne alusta asti.

## 16.6 Tilastot

Taulukko 49: Korttipaikka-projektin tuntitaulukko.

	Esit	Määr	Suun	Tot	Test	As, Ko, Yp	Muu	Yht	%
Tumi	23,5	37,5	22,5	55	8,5	5	3,5	155,5	16,6
Pala	57	56	27,5	95	21	0	0	256,5	27,4
Tark	10	13	6	0	7	0	1	37	4,0
Opet	21	0	7	4	12	0	22	66	7,1
Doku	55,5	55,5	27,5	15	25,5	0	17	196	21,0
Pro/D	0	0	0	0	0	0	0	0	0
Projh	12	5	8	21	6	0	1	53	5,7
Työ	0	6	9,5	122	17,5	5	5,5	165,5	17,7
Yht	183,5	173,5	108	312	97,5	10	50	934,5	100
%	19,6	18,6	11,6	33,4	10,4	1,1	5,4	100	
RE	3	0,5	0	0	0	0	0	3,5	0,4
U-T	1,5	0	0	0	0	0	0	1,5	0,2

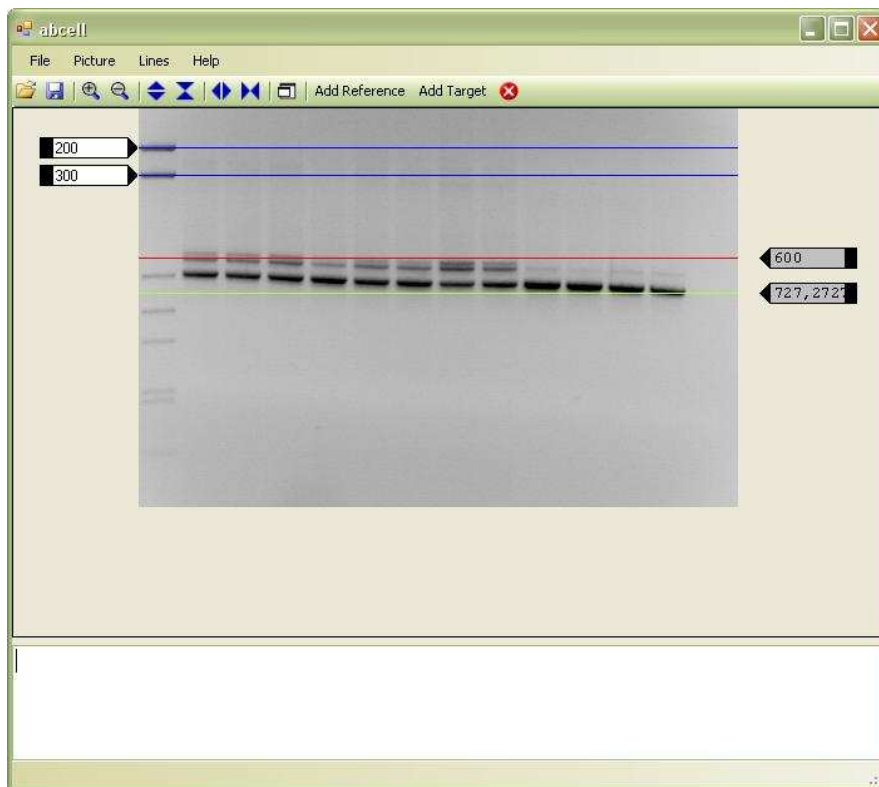
Taulukko 50: Projektin dokumentit.

Dokumentti	sivuja
Projektisuunnitelma	33
Projektin käytettävyyssuunnitelma	18
Vaatimusten määrittely	38
Toteutussuunnitelma	35
Testaussuunnitelma	20
Käyttöohje	20
Käytettävyystestausraportti	6
Loppuraportti	15
Loppukertomus	4
Yhteensä	189

Taulukko 51: Koodirivien lukumäärä.

Ohjelmointikieli	HTML + PHP
LOC	2400
SLOC	2200
Uudelleenkäytetty	0
Uudelleenkäytetty ja muokattu	0
Moduuleja	10
Ohjelmointikieli	XML
LOC	1080
SLOC	1075
Uudelleenkäytetty	0
Uudelleenkäytetty ja muokattu	0
Moduuleja	-
Ohjelmointikieli	Kaikki
LOC	3480
SLOC	3275
Uudelleenkäytetty	0
Uudelleenkäytetty ja muokattu	0
Moduuleja	10

## 17 Visualize DNA



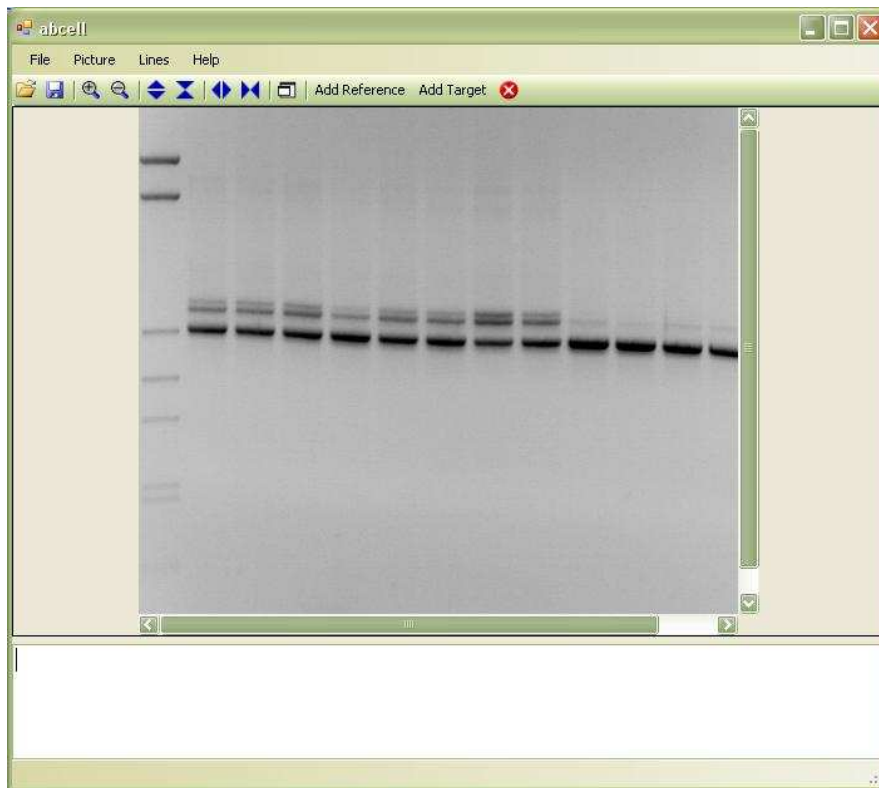
Kuva 23: A screenshot of the program.

### 17.1 Overview

This project is based on the request of Terho Eloranta, the managing director of ABCELL Corporation. This project's task is to implement a software to help biology professionals and researchers compute the location value of molecules in the molecule weighing experiment. This project's goal is to design and implement a software to compute the size value of the molecules on the experiment images, instead of manual calculation. The product of this project will be used by the biology professionals and researchers.

### 17.2 Organization

This chapter describes the project organization and responsibilities in it.



Kuva 24: Another screenshot of the program.

## Project group

The project group was originally structured so that there is one project manager (Jörg Schummer) and 5 team members (Herman Suovaara, Heli Helminen, Rlisa Ryyänen, Musa Sizya and Linfeng Li). Our team was international and project language was english.

The project manager organised the project team's work and intermediates between the supervisor, the client, and the project group. Later, the project manager quit the course without any notice in April. Since that time the group has been organized by itself and also developed the product without a manager. Linfeng Li, Musa Sizya and Herman Suovaara were responsible for programming. Musa and Linfeng were also the math experts for calculation and logarithms. Elisa Ryyänen was responsible for the user interface, usability and testing. Heli Helminen was the risk manager, client contact and concentrated on planning, organizing and documenting. Documentation in general was done by all members in different phases.

Because of these major problems and changes our final product changed

along the way from what it was planned to be. This was inevitable even though all other team members participated during the whole project. Despite of all, we are pretty confident that the software will meet the clients needs.

### **Client**

The client was a company called Abcell Oy, from Tampere, Finland. Abcell Oy is a seller of biotechnological equipment. In the scope of the project, Dr. Terho Eloranta, their managing director, acted on their behalf and fulfilled the client's duties.

### **Others**

Requirements Engineering group made the original requirements. Usability team had influence on the user interface and usability matters. The contact for this project of the usability team was Jaana Huotari.

## **17.3 Problems and risks**

Some risks we couldn't avoid and some risks had high influence on the team success and stress. Risks not foreseen had no new risks, but the foreseen risks were surprisingly likely to happen and had very high influence. We could say that any project is no real if there are no problems.

### **Foreseen risks**

The foreseen risks were divided into six different sections; technology, human, organisational, client, tool and estimation risks. During the whole process problems were openly discussed and we all tried to find a solution for them, and finally did. Major problems occurred because of the failed project management (manager left).

The main technological risk was planning on using programming language (C++) which is not familiar to most project team members. We also had trouble on finding a suitable tool for commercial purposes. The graphics for visualisation and calculation were smaller problems than estimated. Good usability was an important issue and mainly we succeeded. However, there are some problems with usability (some secondary features of the program), which could not be solved in this time and our teams abilities (no professional programmer). These problems are documented for the client. The final product was a bit different from the original requirements, all the major functions were implemented.



The team members were committed to this project and issues were discussed and dealt with. We feel that we had a good team spirit and open conversation. Sometimes it was hard to find a suitable meeting time for everyone and other courses may be "disturbed" the project work. We lost our project manager for an unknown reason. With him we also lost a lot of skills, especially in programming knowledge.

Client is very busy and the meetings with him were therefore very limited and contacting extremely difficult. Luckily the client knew exactly what he wanted. In time we found the mutual language and everything got done (major features). This project was done with minimum (or lower) client activity.

Our major schedule was provided by Timo Poranen, and it dictated the deadlines for each step. Our personal schedule was made by that and we succeeded staying in time with major deadlines despite all the problems.

Tasks and dividing them among the project team was done mostly by Jörg Schummer. Later we worked like "one for all, and all for one" mentality based on voluntariness and abilities of each member. It is quite hard to divide the tasks equally between all team members.

### **Risks not foreseen**

There were actually no risks that were not foreseen. However, student participation, management and reasonable requirements were estimated as their probability to occur is low. In our case they all came true and the seriousness was high as estimated. In high level we knew what to expect, but couldn't see how much and what kind of affects these risks have to the whole project.

## **17.4 Management**

This chapter describes the project management in COM8 project.

### **17.5 Group meetings**

The project team met weekly, sometimes more than once in a week if necessary. Some times we also met as smaller groups, if a task required that people involved in it meet separately to work. Weekly meetings worked really well and we felt them necessary for our project. The main problem was to find a suitable time for everyone. We also communicated a lot by email.

## 17.6 Weekly reports

Weekly reports were written by the project manager Jörg Schummer. Since he resigned, there hasn't been any weekly reports but we informed Timo Poranen on important team matters and progress.

## 17.7 Inspections and reviews

All the required inspections and reviews were held and accepted. There were some small document changes made after feedback but our groups work seemed to be in order.

## 17.8 Methods and Tools

This chapter includes information about used tools and methods used to develop this software.

### Tools

Management tools

- Versioning system (CVS or Subversion)
- Working hour tracking system (Electronic Task/Time Recording, ETTR)
- Project management organizational tool (TaskJuggler)

Development tools:

- Dev-C++ version 4.9.9.2
- wxWidget version 4.6.1
- SharpDevelop
- Microsoft .Net Framework version 2.0

## 17.9 Methods and Techniques

Management methods:

- COCOMO
- Gantt Chart

Design techniques:

- UML

Implementation techniques:

- MVC
- C#

## 17.10 Conclusions

This chapter includes information about group experience in this project.

### Experiences from the project

The project was very interesting but also quite demanding. Our client contact was quite passive and interaction was extremely difficult. His experience in computer and project work was very limited. We had trouble in understanding each other first, but later we found mutual understanding. It was really hard to get any answers and testing material from him, because he didn't respond to our contacts until several reminders.

Very unfortunate setback was losing our project manager. It caused some confusion first but luckily we got everything under control and succeeded to finish the project. We also had to make some major changes to our original requirements with some features required by Jörg. We used too much time to find a free software to make a commercial program. Finally it was too difficult to use and replaced in the end.

We have learned a lot about co-operating with other people, problem solution, international communication, planning. The technical aspect took too much energy from the team. It has been all more like learning about project work as a work style, but unfortunately not so much about programming. We think that this experience was very teaching and found out some aspects of the process in a hard way.

### What to do better next time

All project teams should consist of people with different abilities. All needed talents should be covered, especially programming in order to make a successful project and product. The requirements have to be made by the skills but also to be able to meet the clients needs. It is against good project management to change the requirements and plans so much.

The team spirit has to be good and the project manager is the key person for it. The team should be able to work together for ups and downs and everyone has to be motivated to go all the way despite of problems.

A very passive client causes non-necessary problems. These should be avoided by choosing determined and motivated clients for this course (in real life you can't choose). Co-operation shouldn't take so much energy and delay the tasks on clients behalf.

We had the original requirements from the requirements engineering course. This course had to make their work in a very short time and with quite small amount of information. In order to make the original requirements work better for project team, these two courses should be scheduled together much better.

## 17.11 Statistics

Taulukko 52: Working hour table of project Visualize DNA.

	PREL	REQ	DES	IMPL	TEST	INS,U,MAI	OTHER	Total
PLAN	22.30	13.60	17.00	8.00	4.00		4.00	68.90
MEET	10.50	107.50	37.00	42.00				197.00
INSP	0.50	18.70	5.30	2.50			4.10	31.10
STUDY	43.10	11.10	12.00	8.00	3.00		7.00	84.20
DOCUM	7.00	28.10	44.00	14.00	8.00	6.00	18.50	125.60
PROD/D		2.80	13.00	5.00				20.80
PROJM	11.90	14.90		6.50			2.90	36.20
WORKJ		3		165.00		2.00		170.00
U-Team		2.30						2.30
Total	95.30	202.00	128.30	251.00	15.00	8.00	36.50	736.10

## 18 CALEX2006

### 18.1 Overview

The CALEX2006 project continued the work of the Cook2005 project, which ended in the spring of 2005. The goal of the CALEX2006 project was to update and implement new features into the calendar application known as C2C, which was the end product of Cook2005. The client of CALEX2006 was Nokia Networks, which was the case with Cook2005 as well.

C2C enables users to share availability information stored in the calendar of the phone and arrange meetings with relatives, colleagues and friends through an easy negotiation mechanism. With just a few simple steps users can plan meetings, invite participants and check their availabilities. If required participants are available, the organizer of the meeting can reserve the selected time slot in everyone's calendar.

### 18.2 Organisation and management

The project team of CALEX2006 consisted of six members including the project manager, Henri Sainio. The main developers of the application were Antti Mattila, Johannes Niemelä and Timo Taipalus, while Vesa Huotari and Jouni Mäki-Panula took care of documentation including user interface specification.

### 18.3 Methods and tools

- Weekly project meetings, which turned out to be a bit too time consuming in the end.
- Project meetings via Skype, though experiencing minor technical difficulties from time to time, relieved a lot of time for the actual work process in the end.
- IRC (Internet Relay Chat), which we used to communicate on daily, casual basis.
- Mailing list, which was slightly more formal than IRC, but served as one channel of communication.
- Web site, which served as the central archive for all documentation regarding the project, including working hour observation.

- CVS (Concurrent Versions System), which was essential during the development phase of the project.

## 18.4 Project phases

The milestones of the project can be viewed in the form of notable days.

- 21st of October: The initial meeting of the project team.
- 31st of October: The first meeting of the project team and representatives of the client, Nokia Networks.
- 14th of November: The review of the project plan.
- 18th of November: The review of the project plan with the client followed by a casual evening event.
- 27th of November: An educational day for the project team in order to teach each other aspects required in the project.
- 9th of December: The review of the requirements specification and the initial user interface proposition.
- 14th of December: The presentation of the project and the prototype at the university.
- 8th of January: User interface specification.
- 3rd of February: The review of the test plan.
- 12th of February: Class diagram finished for the design plan.
- 17th of February: The review of the design plan.
- 17th of May: The final presentation of the project at the university.
- 31st of May: The delivery of the final product for Nokia Networks.

## 18.5 Conclusions

Initially the whole point of the project seemed to be the production of a load of documentation in a short period of time. However in February, after the design plan was reviewed, the project changed radically. The work load for those that had the knowledge of taking care of the actual development

process seemed to spun out of control, and there simply were no means to ease this progress.

The last phase of our project seemed to be implementation, meaning we didn't have a separate testing period at all, since the development consumed our resources so extensively.

However, we were quite correct in the initial risk assessment presented in the project plan. We then concluded that the three biggest threats for the success of our project were:

- Lack of experience with the relevant tools and technologies.
- We're unable to estimate the effort that has to put in order to comprehend the work of the previous project.
- Lack of time for the project.

Pretty much all of these emerged in one way or another during the final stages of the project. Over all, the project was slightly too massive for the resources at our disposal.

In our opinion, the course should start earlier in the fall, so that the early phases, which are characterized by the heavy orientation in producing documents, wouldn't require as extensive attention from the entire project team as they did this time.

## 18.6 Statistics

Taulukko 53: Working hour table of project CALEX2006.

	PREL	REQ	DES	IMPL	TEST	INS,U,MAI	OTHER	Total	%
PLAN	57,50	52,50	12	20,75	0	0	0	142,75	10,80
MEET	55	90,50	37	116	2	0	0	300,50	22,74
INSP	0	21	2	1,50	0	0	0	24,50	1,85
STUDY	34	2,50	5,75	4,50	0	0	0	46,75	3,54
DOCUM	24,50	57,25	35,50	27	0	0	0	144,25	10,91
PROD/D	0	0	0	0	0	0	0	0	0
PROJM	8	21,50	6	13,25	2	0	0	50,75	3,84
WORKJ	118,50	115	63	315,75	0	0	0	612,25	46,32
Total	297,50	360,25	161,25	498,75	4	0	0	1321,75	
%	22,51	27,26	12,20	37,73	0,30	0	0		

Taulukko 54: Project's documents.

Document	Pages
Project plan	55
Project's usability plan	0
Requirements specification	28
Implementation plan	24
User interface document	17
Test plan	13
User's guide	5
Test report	0
Maintenance document	14
Final report	19
Final story	3
Total	178

Taulukko 55: Project's codelines.

Ohjelmointikieli	C++
LOC	8056
SLOC	7121
Reused code	0
Reused and modified code	267
Classes	25



## 19 Kuvapankki

Seuraavassa kuvataan Kuvapankki-projektin tilanne 7.5.2006, jolloin projektin toteutus oli toiminnallisilta osiltaan lähes valmis. Tekemättä oli enää joitakin pieniä yksityiskohtia, lopullinen testaus ja dokumenttien viimeistely.

### 19.1 Yleiskuvaus

Projektin tavoitteena oli toteuttaa Loimaan Ammatti-instituutille sovellus jonka avulla olisi mahdollista hallinnoida ja jakaa kuvamateriaalia sekä WWW-käyttöön että muulla tavoin julkaistavaksi. Sovelluksen tuli soveltua instituutin opiskelijoiden ja henkilökunnan, eli yhteensä n. 1000 henkilön, käytettäväksi WWW-käyttöliittymän kautta.

### 19.2 Projektiorganisaatio

Projektiryhmä koostui alunperin kuudesta henkilöstä, mutta yhden ryhmäläisen osalta into loppui kesken jo alkumetreillä, alle 20:n tunnin jälkeen. Seuraava joukko jatkoi projektin loppuun saakka: Pasi Pekkanen, Sari Kinnari, Sirkka Ylander, Samuli Mäki ja Jani Tuisku. Projektin alkuvaiheita edesauttoi se, että Sirkka tunsisi asiakkaan entuudestaan ja oli työssään jo määritellyt vastaavanlaisen järjestelmän. Kellään projektiryhmästä ei ollut kattavaa kokemusta kaikista toteutukseen liittyvistä osa-alueista, joten kaikille tarjoutui tilaisuus oppia jotain uutta.

Pasi toimi projektissa projektipäällikkönä vastaten projektin koordinoimista sekä yhteyksistä asiakkaaseen ja projektin valvojaan (Timo Poranen). Sari vastasi yhdessä Sirkan ja käytettävyyksiin edustajan (Ilari Kajaste) kanssa käyttöliittymän suunnittelusta ja tarvittavien WWW-sivujen toteutuksesta. Sarin ja Sirkan osalle lankesi myös suurin osa projektin dokumentoinnista. Sirkka vastasi tietokannan rakenteen suunnittelusta. Samulin harjoille jäi suurin osa projektin ohjelmoinnista ja testauksesta. Jani toimi palvelimen ylläpitäjänä sekä ohjelmoi ja testasi joitakin oleellisia sovelluksen osia. Kaikki ryhmäläiset osallistuivat ainakin jossain määrin dokumentointiin ja testaukseen, mutta käyttöliittymän suunnittelu- ja toteutusvastuut jakautuivat edellä kuvatulla tavalla hieman epätasaisemmin.

Projektiryhmä kokoontui projektin aikana lähes viikottain. Säännöllinen kokoontuminen mahdollisti ongelmien pohtimisen yhdessä sekä seuraavaan tapaamiseen mennessä suoritettavien tehtävien päättämisen ja jakamisen. Projektiryhmän tapaamisista kirjoitettiin muistiot, jotka asetettiin ryhmäläisten saataville. Projektipäällikkö seurasi projektiryhmäläisten ajankäyttöä

taulukoiden eri työvaiheisiin käytetyt tunnit ja projektiin osallistujien viikoittaisen ajankäytön.

Projektipäällikkö tapasi projektin valvojan joitakin kertoja projektin edetessä, tilanteen niin edellyttäessä. Myös projektiryhmäläiset tapasivat valvojan katselmoitaessa projektin dokumentteja. Asiakkaan kanssa yhteydenpito hoidettiin etupäässä sähköpostin ja puhelimen välityksellä, henkilökohtaisten tapaamisten jäätyä yhteen ainoaan kertaan, jolloin keskusteltiin asiakkaan vaatimukset ja toiveet läpi asiakkaan edustajan (Juha Skyttälä) ja projektiryhmän kesken. Käytettävyyksiin edustaja osallistui projektin tapaamisiin joitakin kertoja projektin suunnitteluvaiheessa.

### 19.3 Välineet, menetelmät ja tekniikat

Loimaan Ammatti-Instituutti hankki projektin käyttöön erityisen Linux-palvelimen, jota hyväksi käyttäen projekti voitiin toteuttaa. Seuraavassa on listattu palvelimen ja sen ohjelmistojen ominaisuudet lyhyesti:

- Palvelin: 2 \* Pentium 4, 3.0 GHz
- Käyttöjärjestelmä: SuSE Linux 9.2 (Kernel 2.6.8-24-smp)
- WWW-palvelinohjelmisto: Apache HTTP Server V2.0.50
- Palvelinsovelluksen ohjelmointikieli: PHP V4.3.8
- Tietokantaohjelmisto: MySQL V4.0.21

### 19.4 Projektin eteneminen

Projekti eteni enimmäkseen suunnitelman mukaisesti, vaikka pieniä esteitä matkan varrella kohdattiinkin. Esteistä kuitenkin selviydyttiin kunnialla eteenpäin. Projektin alkuvaiheissa, projektisuunnitelmaa kirjoitettaessa, projektin tärkeimmiksi riskeiksi arvioitiin seuraavat:

1. kokemattomuus käytetyissä tekniikoissa
2. työmäärä- ja aika-arvion vaikeus
3. kokematon projektiryhmä
4. poisjäänti kesken projektin
5. ajanpuute (liian vähän aikaa projektille)

## 6. asiakas menettää mielenkiinnon

Voidaan sanoa, että tavallaan jokainen näistä riskeistä toteutui asettaen oman haasteensa projektin etenemiselle. Projektiryhmän kokemattomuuden ja ryhmälle uusien toteutustekniikoiden takia jouduttiin opiskelemaan asioita, millä oli vaikutusta projektin etenemiselle. Työmäärä- ja aika-arviot eivät osuneet ihan kohdalleen, mutta tämä lienee odotettavissa projektissa kuin projektissa. Eräs ryhmän jäsenistä jäi pois kesken projektin, minkä johdosta muille kertyi arvioitua enemmän tekemistä. Toisaalta tällä oli positiivinen vaikutus siinä mielessä, että ryhmän koko pieneni ja koordinoinnin tarve väheni samanaikaisesti. Ajanpuute muodostui sikäli ongelmalliseksi, että muiden kiireiden takia ei aina ollut mahdollista panostaa projektityön eteen täysimääräisesti. Asiakkaan mielenkiintokaan ei ollut projektin suhteen ylenpalttinen, mutta tästä seurasi tietty vapaus toteuttaa asioita.

## 19.5 Johtopäätökset projektista

Projektin päällimmäisenä opetuksena on se, että käytettävällä ohjelmistokehitysmenetelmällä on erittäin suuri merkitys projektin edistymisen ja lopputuloksen kannalta. Projektityö-kurssi noudattaa rakenteeltaan perinteistä vesiputousmallia, mutta projektin edetessä kävi selväksi, ettei tämä malli ole välttämättä ollenkaan paras mahdollinen. Kyseisestä mallista on osattava irroittautua mm. uusien asioiden opiskelun suhteen. On nimittäin huomattavan hankalaa, ja monissa tapauksissa jopa mahdotonta, määrittellä asioiden toteuttamista ilman tietoa ja kokemusta siitä, miten eri tekniikat toimivat käytännössä. Kaikesta suunnittelusta ja dokumentoinnista huolimatta olisi siis aloitettava asioiden opiskelu ja toteuttaminen mahdollisimman aikaisessa vaiheessa.

Toinen opittu asia on se, että laajan projektin läpivieminen kurssilaisten muiden opintojen ohessa on haastava tehtävä. Kurssin kestäessä olisi syytä varata aikaa viikkopalaverien lisäksi myös mahdollisimman usein tapahtuville yhdessä opiskelun ja tekemisen jaksoille. Näiden avulla voitaisiin jakaa kokemuksia ja opiskella asioita. Näin pystyttäisiin varmistamaan tasaisempi projektin eteneminen muiden kiireiden ja itsenäisen tekemisen ohessa.

Projekti oli kaikkineen hyvin opettavainen, sillä sen myötä opittiin paljon uusia asioita ja ryhmätyön niksejä tulevien koitosten varalle.

## 19.6 Tilastot

Seuraavissa taulukoissa on esitetty tilastotietoa Kuvapankki-projektista. Ensimmäisessä taulukossa on kuvattuna projektiin käytettyjen tuntien määrä

työvaiheittain ja toiminnoittain. Taulukon tunneista puuttuvat vielä projektin viimeistelyyn tarvittavat tunnit, eikä myöskään kaikkia asioiden opiskeluun käytettyjä tunteja ole kirjattu mukaan.

Taulukko 56: Projektin Kuvapankki tuntitaulukko.

	Esit	Määr	Suun	Tot	Test	As, Ko, Yp	Muu	Yht	%
Tumi	6	8,5	3,5	12	5	2	2	39	5,4
Pala	37	29,5	27	26,5	18	0,5		138,5	19,1
Tark	5	5	2		6			18	2,5
Opet	11	5,5	4	5	3		11	39,5	5,4
Doku	50,5	34,5	29	7	39	0,5	1,5	162	22,3
Pro/D						3	2	5	0,7
Projh	9	10	12,5	17,5	9		4,5	62,5	8,6
Työ		8,5	40	200	6,5	1	1	257	35,4
Yht	122,5	102	118	268	86,5	7	22	726	
%	16,9	14,0	16,3	36,9	11,9	1,0	3,0		

Toisessa taulukossa on esitetty projektin kuluessa syntyneiden dokumenttien sivumäärät. Vielä kirjoittamattomien dokumenttien osalta taulukossa on sivumäärän kohdalla viiva.

Taulukko 57: Projektin dokumentit.

Dokumentti	sivuja
Projektisuunnitelma	36
Vaatimusten määrittely	33
Toteutussuunnitelma	30
Testaussuunnitelma	32
Käyttöohje	-
Testausraportti	-
Ylläpito-ohje	-
Loppuraportti	16
Loppukertomus	4
Yhteensä	151+

Kolmannessa taulukossa on esitetty tilastotietoja projektin toteutukseen käytetyn PHP-koodin osalta.

Taulukko 58: Koodirivien lukumäärä.

Ohjelmointikieli	PHP
LOC	4130
SLOC	n. 3750
Uudelleenkäytetty	n. 20
Uudelleenkäytetty ja muokattu	n. 100
Tiedostoja (PHP)	27

## **20 Safety At Work**

### **20.1 Yleiskuvaus ohjelmasta**

Safety at Work -projektin tehtävänä oli tuottaa Koulutuskeskus Salpaukselle Moodle-pohjainen verkko-oppimisympäristö, jossa opiskelija voi itsenäisesti opiskella teollisuudessa käytössä olevan työturvallisuuskortin saamiseksi vaadittuja tietoja. Opiskelijalle tarjotaan oppimisympäristössä monipuolista opiskelumateriaalia työturvallisuuskoulutuksen eri osa-alueista. Opiskeltava materiaali on jaettu viiteen eri osa-alueeseen ja opiskelija voi opiskella ja käydä osioita läpi haluamassaan järjestyksessä niin kauan kuin katsoo tarpeelliseksi. Verkko-oppimisympäristössä otetaan huomioon, että opiskelijat suorittavat koulutuksen hyvin erilaisilla taustatiedoilla, joten opiskelijan tulee voida itse päättää, mitä opiskelee ja kuinka paljon. Jokaisen eri osa-alueen hallintaa testataan pienillä harjoituksilla ja yhdellä laajemmalla loppuharjoituksella. Tällä hetkellä tarjolla ei ole vielä mahdollisuutta suorittaa koko työturvallisuuskorttikoulutusta verkossa, eli varsinainen kortin saamiseen tarvittava tentti tehdään vielä verkon ulkopuolella.

### **20.2 Projektiorganisaatio**

Ryhmän projektipäällikkönä toimi Ville Parviainen. Projektipäällikön vastuulla oli mm. aikataulutus, työnjako ja yleinen projektin etenemisen seuranta ja hallinta. Vastuualueet projektiryhmän sisällä jaettiin pääpiirteissään siten, että Antti Syrjä ja Bemmu Sepponen vastasivat teknisestä toteutuksesta ja koodauksesta, Johanna Hakola ja käytettävyyksiin jäsen Suvi Peltomäki käyttöliittymäsuunnittelusta ja -testauksesta, Timi Antere testauksesta ja osin koodauksesta ja Jari Hautamäki ohjelman vaatimuksista, asiakkaan konsultoinnista ja pedagogisten näkökulmien huomioimisesta.

### **20.3 Välineet, menetelmät ja tekniikat**

Oppimisympäristö on rakennettu Moodle-kurssijärjestelmän päälle ja se käyttää MySQL-tietokantaa. Oppimisympäristö on laadittu niin, että se toimii millä tahansa laitteistolla, jossa myös Moodle-kurssijärjestelmä ja MySQL toimivat. Toteutuksessa pystyttiin käyttämään melko paljon Moodle-ympäristössä jo olevia ominaisuuksia, joten koodausta ei tarvinnut aloittaa kokonaan alusta. Itse Moodle-kurssijärjestelmä on toteutettu PHP-kielellä.

## 20.4 Projektin eteneminen

Projekti toteutettiin vesiputousmallin mukaisesti ja se sopi tälle projektille melko hyvin. Projektityökurssille oli annettu valmis aikataulutusta, jonka puitteissa edettiin. Syksyllä pääpaino oli projektin yleisessä suunnittelussa ja sovelluksen määrittelyssä ja joulukuun mennessä piti olla valmiina pieni demo tulevasta sovelluksesta. Kevät aloitettiin käyttöliittymän suunnittelulla ja prototyyppien testailulla. Varsinaista sovellusta koodattiin maaliskuussa. Huhtikuun loppupuolella suoritimme käytettävyydestä kolmen kohde-ryhmään sopivan testihenkilön avulla. Testipäivän jälkeen korjasimme vielä testissä ilmenneitä pahimpia ongelmakohtia ja mietimme, mitä jatkokehittävää asiakkaalle vielä jää.

## 20.5 Johtopäätökset

Projekti onnistui varsin hyvin, pysyimme aikataulussa ja oppimisympäristö on sellainen kuin pitikin. Tosin etenkin toteutuksessa ja käytettävyydestä tuli hieman kiire, mutta missään vaiheessa työ ei kuitenkaan tuntunut ylivoimaiselta. Joitain määrittelyvaiheessa tehtäväksi suunniteltuja (matalan prioriteetin) ominaisuuksia jäi toteuttamatta, mutta kaikki tärkeimmät toiminnot saatiin toteutettua. Käytettävyydestä tuli myös ilmi jatkokehityksen kohteita, mutta kuten sanottu, se tehtiin mitä aiottiinkin.

Asiakas oli tyytyväinen tehtyyn sovellukseen ja aikoo jatkokehittää tuotetta ja ottaa oppimisympäristön käyttöön jossain vaiheessa.

## 20.6 Tilastot ja kuvia sovelluksesta

Taulukko 59: Dokumenttien sivumäärät

Dokumentti	Sivumäärä
Projektisuunnitelma	16
Käytettävyyssuunnitelma	14
Vaatimusmäärittely	17
Toteutussuunnitelma	8
Käyttöliittymäsuunnitelma	15
Testaussuunnitelma	25
Käytettävyystestiraportti	12
Loppukertomus	4
Kokonaissivumäärä	111

**Työtuntikirjanpito - Kaikki työtunnit yhteensä**

	Esituk	Maarit	Suunnit	Toteutus	Testaus	As, Ko, Yp	Muu	Yhteensä
Tumi	16:30	9	3	20	3:30			52
Pala	35	65:30	29	40:30	17			187
Tark	4	6	4		4			18
Opet	18	16	11	6	9		2	62
Doku	35	38	26	46	5:15	5	2	157:15
Pro/D	2	3	1	3				9
Projh	41:30	8	19	28	9	4	7	116:30
Työ		2	18	84	25			112:45
Yhteensä	152	147:30	111	227:30	72:45	9	123:45	843:30

Kuva 25: Safety at Work -projektin tunnit 11.05.2006 mennessä.



KOULUTUSKESKUS  
**SALPAUS**

**Työturvallisuuskortti**

Et ole kirjautunut. (Kirjaudu)

**Kurssi**

[Etusivulle](#)  
[Keskustelualue](#)

**Oppimateriaali**

- Työturvallisuuden perustekijät
- Tapaturmien ennakointi
- Työturvallisuusohjeita
- Keskeiset vaaratekijät ja niiden torjunta
- Toiminta onnettomuusilanteessa
- Loppuharjoitus

**Omat tiedot**

[Omien tietojen muokkaus](#)  
[Etenemisen seuranta](#)

[Ohjeet](#)

**TYÖTURVALLISUUSKORTTI**

Työturvallisuuden kehittämiseksi teollisuuden yhteisillä työpaikoilla on kehitetty valtakunnallinen menettelytapa, josta käytetään nimeä työturvallisuuskorttikoulutus.

Työturvallisuuskortin käyttöönotto ei ole yrityksille pakollista, mutta useat yritykset tulevat kuitenkin vuoden 2005 alusta edellyttämään lihanhankkijoidensa työntekijöiltä työturvallisuuskorttia. Korttikoulutus ei korvaa työpaikkakohtaista perehdyttämistä vaan antaa perusosaamisen sen omaksumiseen.

Kurssin hyväksyttävästä suorittamisesta myönnetään työturvallisuuskortti, joka on voimassa viisi vuotta. Kurssin toteutuksesta vastaa turvallisuuskortin kouluttajakoulutuksen suorittanut kurssinjohtaja.

**Kurssin suorittaneella on**

- \* perustiedot työympäristön vaarojen tunnistamiseen
- \* tiedot työsuojelun yleisistä periaatteista
- \* tiedot yhteistoiminnasta yhteisillä työpaikoilla
- \* perusvalmiudet omaksua työpaikkakohtainen ja työtehtäväkohtainen perehdyttäminen

Kuva 26: Etusivu.

KOULUTUSKESKUS SALPAUS Työturvallisuuskortti

Olet kirjautunut nimellä Testi Testaaja. (Kirjaudu ulos)

» Etusivulle

### Loppuharjoitus - Yritys 4

**1** Työturvallisuutta parantava toimenpide tehdään ainoastaan jos todennäköisyys sen sattumiselle on yli 50 %.

Vastaus:  Tosi  Epätosi

**2** Vastuu yksittäisen työtehtävän työturvallisuudesta ei ole kenellekään.

Vastaus:  Tosi  Epätosi

**3** Kun odottamaton käynnistyminen estetään lukitsemalla turvakytin, käynnistyskytkimeen ei enää tarvitse laittaa käynnistyskieltoilpiä.

Vastaus:  Tosi  Epätosi

Kuva 27: Loppuharjoitus.

KOULUTUSKESKUS SALPAUS Työturvallisuuskortti

Olet kirjautunut nimellä Testi Testaaja. (Kirjaudu ulos)

» Etusivulle

### Safety at work etenemisen seuranta: Testi Testaaja

	Osio	Luvut	Harjoitus
✓	1. Työturvallisuuden perustekijät	1.1 1.2 1.3 1.4 1.5 1.6	3/3
✓	2. Tapaturmien ennakointi	2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8	3/3
✓	3. Työturvallisuusohjeita	3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8	3/3
	4. Keskeiset vaaratekijät ja niiden torjunta	4.1 4.2 4.3 4.4 4.5 4.6 4.7 4.8	
		4.9 4.10 4.11 4.12 4.13 4.14 4.15 4.16	2/3
	5. Toiminta onnettomuusilanteessa	5.1 5.2 5.3 5.4 5.5 5.6 5.7 5.8	0/3

Täysin pistein suoritettut harjoitukset sekä luvut, joita on luettu, näkyvät vihreällä taustaväriillä.

Kuva 28: Etenemisen seuranta.