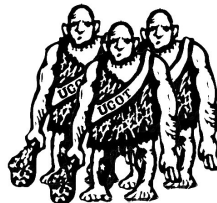


UCOT-Sovellusprojekti

Projektiraportti

Ilari Liukko
Tuomo Pieniluoma
Vesa Pikki
Panu Suominen



Versio: 0.04
Julkinen
18. joulukuuta 2006

Jyväskylän yliopisto

Tietotekniikan laitos

Jyväskylä

Hyväksyjä	Päivämäärä	Allekirjoitus	Nimenselvennys
Projektipäällikkö	__.__.2006		
Tilaaaja	__.__.2006		
Ohjaaja	__.__.2006		

Tietoa dokumentista

Tekijät:

- | | | |
|-------------------------|----------------------|-------------|
| • Ilari Liukko (IL) | ilanliuk@cc.jyu.fi | 050-4367494 |
| • Tuomo Pieniluoma (TP) | tujupien@cc.jyu.fi | 040-7202054 |
| • Vesa Pikki (VP) | vevijopi@cc.jyu.fi | 044-5288031 |
| • Panu Suominen (PS) | panu.suominen@iki.fi | 050-3458484 |

Dokumentin nimi: UCOT-projekti, Projektiraportti

Sivumäärä: 40

Tiedosto: UCOT-projektiraportti-0.04.tex

Tiivistelmä: Tämä dokumentti on raportti UCOT-projektin läpiviennistä. Dokumentissa kuvataan projektin tavoitteiden toteutumista ja sitä, kuinka se vastasi projektisuunnitelmaa.

Avainsanat: UCOT, heuristiikka, käyttötapaus, olioluokka.

Versiohistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.01	1.12.2006	Luonnoksen tekeminen aloitettu.	VP
0.02	11.12.2006	Luvut 1-3 kirjoitettu.	VP
0.03	12.12.2006	Luvut 4-8 kirjoitettu ja Villen huomauttamattomat kohdat korjattu	VP,TP
0.04	18.12.2006	Katselmoinnissa löytyneet virheet korjattu	VP

Tietoa projektista

UCOT-projekti suunnitteli ja toteutti Jyväskylän yliopiston tietotekniikan laitokselle ohjelmiston. Tällä ohjelmistolla voidaan heuristiikkoja käyttäen etsiä käyttötapauskista ohjelmistokehityksen analyysivaiheen olioluokkia.

Tekijät:

- | | | |
|-------------------------|----------------------|-------------|
| • Ilari Liukko (IL) | ilanliuk@cc.jyu.fi | 050-4367494 |
| • Tuomo Pieniluoma (TP) | tujupien@cc.jyu.fi | 040-7202054 |
| • Vesa Pikki (VP) | vevijopi@cc.jyu.fi | 044-5288031 |
| • Panu Suominen (PS) | panu.suominen@iki.fi | 050-3458484 |

Tilaaaja:

- | | | |
|--------------------|--------------------|-------------|
| • Tommi Kärkkäinen | tka@mit.jyu.fi | 040-5677854 |
| • Antti Hakala | anthakal@cc.jyu.fi | 040-7096224 |

Ohjaajat:

- | | | |
|---------------------|--------------------|-------------|
| • Ville Isomöttönen | vilisom@cc.jyu.fi | 014-2604976 |
| • Miika Nurminen | minurmin@cc.jyu.fi | 014-2602530 |

Tarkkailija:

- | | | |
|-------------------|--------------------|-------------|
| • Antti Hallamäki | antahall@cc.jyu.fi | 044-3555356 |
|-------------------|--------------------|-------------|

Yhteystiedot:

- | | |
|---------------------|---|
| • Sähköpostilistat: | ucot06@korppi.jyu.fi,
ucot_opetus@korppi.jyu.fi |
| • Projektiarkisto: | https://korppi.jyu.fi/list-archive/ucot06/ind.html |
| • Opetusarkisto: | https://korppi.jyu.fi/list-archive/ucot_opetus/ind.html |
| • Työhuone: | Ag C222.2 / 014-2604963 |

Sisältö

1	Johdanto	1
1.1	Dokumentin tarkoitus	1
2	Tavoitteiden toteutuminen	3
2.1	Sovellus	3
2.2	Projektiryhmän oppimistavoitteet	3
3	Organisaatio ja resurssit	5
3.1	Koulutus	5
3.2	Resurssit	6
4	Hallintatavat	7
4.1	Hallinta	7
4.2	Palaverit	8
4.3	Dokumentit ja katselmoinnit	8
4.4	Versionhallinta	9
4.5	Tiedotus	9
5	Tehtävät, työmäärät ja työnjako	10
5.1	Ryhmän työnjako	11
5.1.1	Ilari Liukko	12
5.1.2	Tuomo Pieniluoma	13
5.1.3	Vesa Pikki	14
5.1.4	Panu Suominen	15
6	Prosessimalli ja aikataulu	16
6.1	Prosessimalli	16
6.2	Iteraatioiden ajankäyttö	16
6.3	Viikkotuntien jakautuminen	20
6.3.1	Ilari Liukko	21
6.3.2	Tuomo Pieniluoma	22
6.3.3	Vesa Pikki	23
6.3.4	Panu Suominen	24
7	Riskit ja niiden seuranta	25
7.1	Henkilöihin liittyvät riskit	26

UCOT-projekti	Projektiraportti 0.04	Julkinen
7.2	Toteutukseen liittyvät riskit	26
7.3	Sovellusaluekohtaiset riskit	27
8	Kokemuksia projektista	28
8.1	Ilari Liukko	28
8.2	Tuomo Pieniluoma	28
8.3	Vesa Pikki	29
8.4	Panu Suominen	29
9	Lähteet	30
Liitteet		
A	Iteraatioiden ajankäytön suunnitelmat	31
A.1	Ensimmäinen iteraatio	31
A.2	Toinen iteraatio	32
A.3	Kolmas iteraatio	33
A.4	Neljäs iteraatio	34
A.5	Viides iteraatio	35
A.6	Asiakastapaamiset	36
B	Termit	37

1 Johdanto

UCOT-projekti toteutti Jyväskylän yliopiston tietotekniikan laitokselle ja Agora Centerille ohjelmiston, jolla analyysivaiheen olioluokkia voidaan muodostaa käyttötapauskuvauksista. Sovelluksen tarkoituksena on osittain automatisoida analyysivaiheen olioluokkien suunnittelua ja osoittaa tuloksen automatisoinnin mielekkyys (proof of concept). Ohjelmisto toteutettiin Tekes-hankkeen rahoittamalle Tuotanto 2010 -tutkimusprojektille.

Oliokeskeisen ohjelmistotuotannon kaksi ensimmäistä vaihetta ovat vaatimusmäärittely ja -analyysi. Vaatimusmäärittelyvaiheessa asiakas ja järjestelmää kehittävä osapuoli määrittelevät järjestelmän tarkoituksen ja analysoivat ongelman, jota ratkaisemaan järjestelmä kehitetään. Tuloksena on vaatimusmäärittely ja myös käyttötapauskuvaus, mikäli käytetään käyttötapauspohjaista kehitysmenetelmää. Käyttötapauskuvaus voidaan löytää kirjoitettua tekstiä tutkimalla tärkeimmät sovellusalueen käsitteet, joista tehdään analyysivaiheen olioluokkia. Analyysivaiheen tarkoituksena on tuottaa sovellusaluetta kuvaava malli, jonka pohjalta järjestelmä voidaan suunnitella ja toteuttaa.

Projekti toteutettiin neljän hengen ryhmässä. Ryhmään kuuluivat Ilari Liukko, Tuomo Pieniluoma, Vesa Pikki ja Panu Suominen. Projektipäällikkönä toimi Vesa Pikki. Projektin vastaavana ohjaajana toimi Ville Isomöttönen ja teknisenä ohjaajana Miika Nurminen. Asiakkaan edustajina toimivat Tommi Kärkkäinen ja Antti Hakala.

Projekti toteutettiin ketterällä prosessimallilla kuudella 2-3 viikon iteraatioilla, joista ensimmäinen alkoi 22.9. ja viimeinen loppui 20.12.

1.1 Dokumentin tarkoitus

Tämä dokumentti on raportti UCOT-projektin läpiviennistä. Dokumentissa kuvataan projektin tavoitteiden toteutumista ja sitä, kuinka se vastasi projektisuunnitelmaa [1]. Näiden lisäksi ryhmän jäsenet kertovat kokemuksistaan projektista.

Luvussa 2 esitellään projektin tavoitteiden toteutumista. Luvussa 3 analysoidaan, kuinka projekti hyödynsi käytössä olevia resursseja. Luvussa 4 tutkitaan projektin hallintatapoja ja niiden onnistumista projektin läpiviennin aikana. Luvussa 5 analysoidaan iteraatioiden ajankäytön toteutumista, sekä tarkastellaan, toteutuivatko

suunnitellut työtehtävät. Luvussa 6 pohditaan projektissa käytettyä prosessimallia ja sen suunniteltua aikataulua, sekä sitä miten ne soveltuivat ohjelmiston toteutukseen. Luvussa 7 analysoidaan projektin riskien toteutumista. Luvussa 8 ryhmän jäsenet kertovat kokemuksistaan sekä siitä, mitä ovat oppineet projektista.

2 Tavoitteiden toteutuminen

Tässä luvussa kuvataan projektin tavoitteiden toteutumista.

2.1 Sovellus

Projektin tavoitteena oli toteuttaa analyysivaiheen oliomalleja käyttötapauksista tuottava sovellus. Sovellukseen tuli myös toteuttaa käyttöliittymä, jolla kyseistä oliomallia voidaan muokata. Sovelluksen tavoitteet ja niiden toteutuminen on listattu tarkemmin projektin vaatimusmäärittelyssä [3]. Sovelluksen vaatimusten toteutus-tapa selviää projektin sovellusraportista [2].

UCOT-projekti toteutti kaikki olennaisimmat vaatimukset. Pää tavoitteena oli toteutuskelpoisuuden todistaminen, joka toteutui jo toisen iteraation kohdalla. Toteutulla sovelluksella voidaan käyttötapauksia ladata, jäsentää ja heuristiikkoja soveltamalla muuttaa ne olimalleiksi. Sovelluksella voidaan myös muokata oliomallia ja tallentaa se eri tiedostomuodoissa.

Ensimmäistä iteraatiota lukuunottamatta jokaisessa iteraatiossa oli aina muutama toteutumatta jäänyt vaatimus. Toteutumatta jääneistä vaatimuksista jokaisesta oli kuitenkin aina toiminnallisuus ajallaan tehty ja vaatimuksen toteutumisen esti bugi. Näistä huolimatta projekti pysyi kuitenkin aina aikataulussa ja kaikki tärkeät vaatimukset toteutuivat.

2.2 Projektiryhmän oppimistavoitteet

Projektiryhmän päätavoitteet olivat saada käytännön kokemusta projekti- ja ryhmätyöstä, sekä oppia käytettävä prosessimalli. Vaikka projektiryhmäläisillä ei ollut aikaisempaa kokemusta projektityöskentelystä onnistui projektimuotoinen ryhmätyöskentely ongelmitta. Projektiryhmän jäsenet oppivat parantamaan keskinäistä ja ryhmän ulkoista kommunikaatiota alkukankeuksista huolimatta. Tätä edisti se, että projektiryhmän jäsenet näkivät toisensa päivittäin ja sähköpostilistaa hyödynnettiin aktiivisesti. Tehtävien jakaminen ryhmän sisällä toimi hyvin ja ryhmän jäsenet oppivat luottamaan toistensa taitoihin. Yleinen henki projektin läpiviennin ajan oli positiivinen.

Projekti toteutettiin käyttäen ketterää prosessimallia. Se tuli erittäin tutuksi projekti-ryhmäläisille, sillä jokainen projektiryhmän jäsen osallistui kaikkiin ohjelmistoprojektin vaiheisiin ja tehtäviin. Projektipäällikön tehtäviä hoiti kuitenkin vain Vesa Pikki, joka toimi projektipäällikkönä koko projektin läpiviennin ajan. Painotus eri tehtäviin jakautui projektin alun tehtävien mukaisesti. Esimerkiksi Panu Suominen teki 0-iteraation aikana testin Abbottin heuristiikan implementoimisesta ja hänen vastuulle päättyi lopulta sovelluksen heuristiikan toiminta. Dokumenttien kirjoituksesta jokaisella oli vastuullaan ainakin yksi isompi dokumentti. Vastuusta huolimatta dokumentit toteutettiin aina pareittain. Dokumenttien kirjoituksessa ryhmä oppi katselmointikäytänteitä ja myös parantamaan omia kirjoitustaitojaan.

Yleisesti projektin aikana ryhmä oppi lisäksi suunnittelemaan projektin läpivientiä, sekä sovelluskehityksen aikataulun suunnittelua. Ryhmän taidot kehittyivät myös arkkitehtuurin suunnittelussa. Ryhmä oppi myös toisiltaan paljon ohjelmoinnista. Aihealueen takia myös käyttötapaukset ja analyysivaiheen oliosuunnittelu tulivat tutummaksi.

3 Organisaatio ja resurssit

Projektiryhmällä oli projektisuunnitelmassa [1] määritellyt resurssit ja koulutus käytössään. Tässä luvussa kuvataan kuinka projektiryhmä hyödynsi niitä.

3.1 Koulutus

Projektiryhmä sai sovellusprojektiin kuuluvaa koulutusta, johon kuului seuraavat luennot:

- Projektin johtaminen ja hallinta
- CVS
- Käytettävyys
- Tekijänoikeus ja sopimukset
- Väliesittelyt

Projektiryhmä koki luennot osin hyödylliseksi, mutta hieman liian pitkiksi ja tasoltaan liian alhaiseksi suunnatuiksi. Ne järjestettiin kesken projektin läpiviennin ja olivat suhteellisen pitkiä. Tällöin ne keskeyttivät työskentelyn ja usein projektiryhmä ei enää luentojen jälkeen jaksanut jatkaa normaalia työskentelyä. Monilla luennoilla käytiin myös läpi asioita, jotka ovat tulleet esille muiden opintojen yhteydessä, varsinkin käytettävyysluennoilla, eikä projektiryhmä kokenut kuin osan kulutetusta ajasta hyödylliseksi. Projektiryhmän mielestä vastaisuudessa oheislunnnot tulisi järjestää ennen projektin alkua tai projektin alussa väliesittelyjä lukuunottamatta.

Projektiryhmän ehdotus käytettävyyspäivän parantamiseksi olisi tiivistää luento lyhyemmäksi karsien muilla kursseilla ilmi tulleet asiat. Tämän lisäksi käytettävyysluennoitsija kiertäisi projektiryhmien luona ja auttaisi projektiryhmiä parantamaan sovellusten käytettävyyttä henkilökohtaisesti. Tällöin jokainen ryhmä saisi sovelukselleen sopivaa koulutusta ja ryhmät, joilla ei ole käyttöliittymää eivät suotta joudu pitkille luennoille, joista ei projektin läpiviennin kannalta ole hyötyä.

Projektiryhmä itse toivoi Bugzilla-koulutusta, mutta totesi Bugzillan olevan tarpeettoman raskas projektin laajuuteen suhteutettuna.

3.2 Resurssit

Projektiryhmällä oli käytössään projektitila ja laitoksen tarjoamat neljä tietokonetta. Projektiryhmä toteuttikin sovelluksen ja dokumentit projektitilassa laitoksen tarjoamilla tietokoneilla. ATK-tuki asensi projektiryhmän pyynnöstä kahteen projektitilan tietokoneeseen Windows-käyttöjärjestelmän, sillä projektiryhmä koki sen parantavan tehokkuutta. ATK-tuki asensi projektiryhmän pyynnöstä projektitilan koneisiin tarvittavat ohjelmistot, sekä tarjosi tukea erinäisissä vikatilanteissa.

Projektin läpiviennin loppuvaiheessa projektiryhmä halusi myös varmistaa sovelluksen Linux-yhteensopivuuden. Tätä varten ATK-tuelta toimitti erillisen Linux-kone testikäyttöön. Tätä konetta Vesa Pikki käytti viikon sovelluksen kehittämiseen ja näin varmisti sen toimivuuden Linux-ympäristössä.

4 Hallintatavat

Tässä luvussa kerrotaan, kuinka projektin etenemistä hallittiin tarkastuspisteiden, palaverien ja ajankäytön raportoinnin avulla.

4.1 Hallinta

Projektin etenemistä seurattiin iteraatioiden taitoskohtiin sijoittuvissa palavereissa. Niissä käytiin formaalisti läpi toteutetut vaatimukset ja päätettiin seuraavan iteraation tehtävistä. Iteraatioiden väliin sijoittuvissa vapaamuotoisissa tarkastuspisteissä keskusteltiin palavereissa epäselväksi tai kesken jääneistä asioista. Tarkastuspistekäytäntö todettiin erittäin hyväksi käytettävässä prosessimallissa. Tällöin projektiryhmä pääsi kasvottain keskustelemaan asiakkaan edustajien kanssa viikottain paria poikkeusta lukuunottamatta.

Projektin palavereissa asiakkaan edustajien annettiin käyttää sovellusta, kun käytiin läpi toteutettuja vaatimuksia. Tämä todettiin hyväksi ratkaisuksi, sillä asiakkaan edustajat löysivät aina bugeja. Tällöin projektiryhmä sai myös ensiarvoista kokemusta käyttöliittymän toimivuudesta ja intuitiivisuudesta. Usein asiakkaan edustajan käyttäessä sovellusta heräsi paljon keskustelua sovelluksesta sekä mahdollisista kehitysideoista.

Projektisuunnitelmassa [1] suunniteltuja projektiryhmän sisäisiä palavereja ei katsottu tarpeelliseksi, sillä projektiryhmän projektitila avoimuutensa ja tilansa puolesta kannusti keskinäiseen kommunikointiin. Tämän lisäksi projektipäällikkö jokaisen päivän lopuksi lyhyesti keskusteli jokaisen projektiryhmäläisen päivän työn tuloksista. Tällöin projektipäällikkö oli tietoinen projektin etenemisestä.

Projektitilassa oli myös valkotaulu, johon jokaisen iteraation taitoskohdassa kirjoitettiin iteraation tehtävät ja niiden vastuuhenkilöt. Taulusta ilmeni myös tehtävien prioriteetti ja toteutumisprosentti. Tämä myös auttoi projektipäällikköä seuraamaan projektin etenemistä, sekä projektiryhmän jäsenet pystyivät siitä itsenäisesti valitsemaan tehtäviä prioriteetin ja vastuun perusteella. Tämä käytäntö todettiin erittäin hyväksi ja toimivaksi.

Projektiryhmällä oli käytössä teknistä ja projektin läpivientiin liittyvää ohjausta, jota pyydettiin ongelmatilanteiden ja kysymysten ilmaantuessa.

Projektiryhmän ajankäyttöä hallittiin Excel-ajankäyttösovelluksella, johon työtunnit ja tehtävät kirjattiin projektin läpiviennin ajan. Sillä pystyttiin seuraamaan suunniteltujen työtuntien toteutumista, sekä käyttää sitä apuna seuraavien tehtävien suunnittelussa.

Projektin dokumentit ja muut tuotokset sijoitettiin projektin WWW-sivuille. Dokumentteista sijoitettiin kaikki eri versiot WWW-sivuille asiakkaan toivomuksesta. Projektin WWW-sivuilta löytyvät seuraavat dokumentit ja tuotokset:

- Projektiin liittyvät dokumentit, jotka kuvattiin projektisuunnitelmassa [1]
- Palaverien esityslistat ja pöytäkirjat
- Ajankäytön raportointi, joka sisältää kuvauksen projektin ryhmän jäsenten käyttämistä työtunneista
- Sovelluksen lähdekoodi
- Projektiryhmän tuottamat muut dokumentit

4.2 Palaverit

Palaverien puheenjohtajan ja sihteerin tehtävät kiersivät projektin ajan jäseneltä toiselle projektisuunnitelmassa [1] suunnitellulla tavalla yhtä poikkeusta lukuunottamatta. Puheenjohtajan tehtävänä oli lähettää esityslista palaveriin osallistujalle ja toimia palaverissa puheenjohtajana. Sihteerin tehtävänä oli laatia pöytäkirja palaverista, lähettää se palaverin jälkeen asianomaisille, kirjoittaa katselmoitaville dokumenteille ehdotetut muutokset ylös, sekä kirjoittaa iteraation tuloksia tarkastellessa esiintyneistä virheistä virheraportti.

Projektin palavereista on koottu taulukko joka on liitteenä A.6.

4.3 Dokumentit ja katselmoinnit

Projektin dokumentit ladottiin L^AT_EX 2_ε-ladontaohjelmalla käyttäen laitoksen tarjoamaa dokumenttipohjaa. Dokumentit julkaistiin PDF-muodossa projektin WWW-sivuilla. Tiedostojen nimeämismuoto on kuvattu tarkemmin dokumentissä [1].

Dokumenttien katselmointi tapahtui palaverissa asennus- ja käyttöohjetta lukuunnottamatta. Katselmointi alkoi sillä, että projektiryhmä lähetti dokumentin kaksi työpäivää ennen katselmointia projektin sähköpostilistalle. Palaverin osapuolet lukivat dokumentin ennen palaveria ja valmistautuivat palaverissa esittämään huomioita dokumentista. Katselmointia palaverissa johti dokumentista vastaava ryhmän jäsen. Katselmoinnissa dokumentti käytiin läpi sivu sivulta sihteerin kirjatessa dokumentille ehdotetut muutokset. Lopuksi dokumentille asetettiin tarkastajat, jotka tarkistivat dokumentille ehdotettujen muutosten toteutumisen. Heidän hyväksytyä toteutetut muutokset, voitiin dokumentti hyväksyä.

4.4 Versionhallinta

CVS pohjainen versionhallinta otettiin projektissa käyttöön heti CVS-luennon jälkeen. Se todettiin erittäin hyväksi ja käytännölliseksi ratkaisuksi projektin tiedostojen jakamiseen, varmuuskopioimiseen ja version hallintaan.

4.5 Tiedotus

Tiedottaminen tapahtui pääosin sähköpostitse ja palaverissa. Projektin tiedotuksesta vastasi projektipäällikkö. Tämä mahdollisti sen, että muut projektiryhmän jäsenet pystyivät keskittymään käsillä oleviin tehtäviin. Ongelmatilanteista tiedotettiin välittömästi sähköpostilistalle.

5 Tehtävät, työmäärät ja työnjako

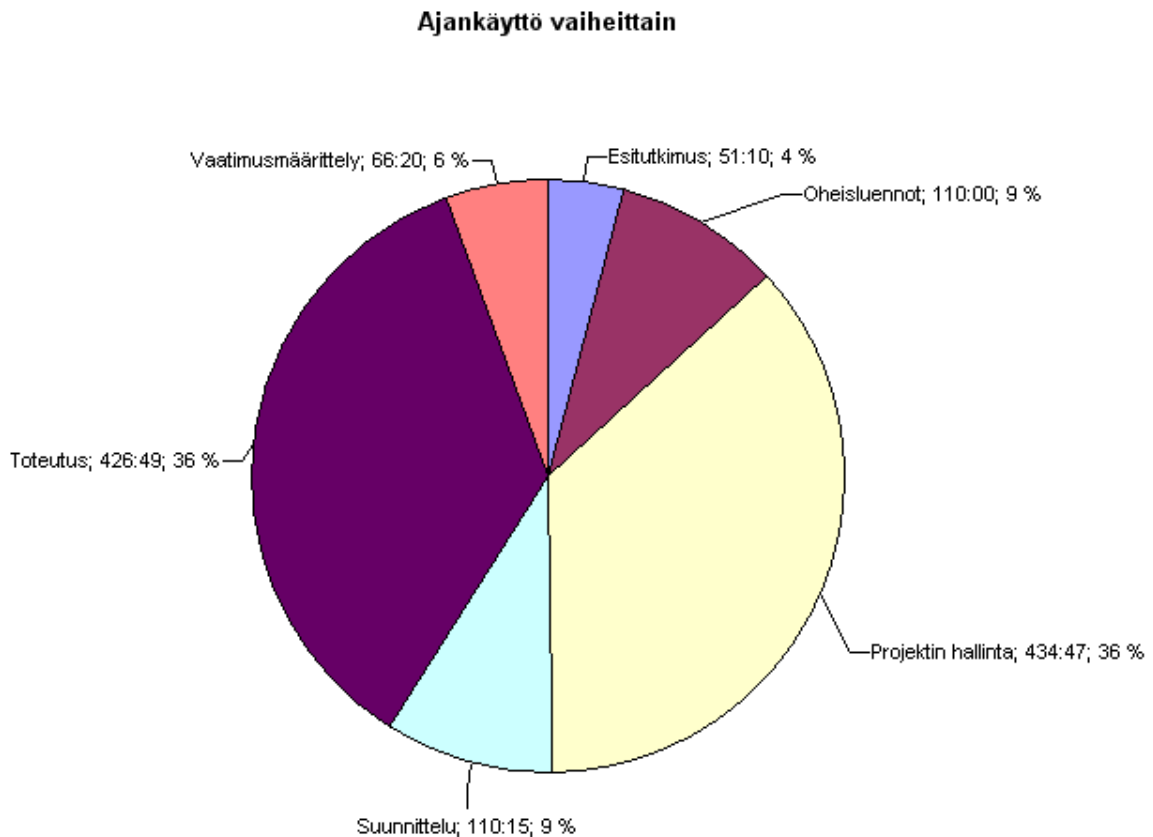
Tässä luvussa tarkastellaan työmäärien ja tehtävien jakautumista projektiryhmän kesken. Työtehtävät suunniteltiin iteraatiokohtaisesti, tästä kerrotaan tarkemmin luvussa 6.

Työtehtäviä ovat esitutkimus, oheisluennot, projektin hallinta, suunnittelu, toteutus ja vaatimusmäärittely. Esitutkimukseen kuuluvat tekniikoiden ja toteutustapojen tutkiminen, kuten esimerkiksi GXL-tiedostomuoto. Oheisluento-työtehtäviin kuuluvat sovellusprojektin oheiskurssin sisältämät luennot, esittelyt ja esittelyihin valmistautuminen. Projektin hallintaan kuuluvat seuranta ja tiedotus, palaverit ja niihin liittyvät toimet, ajankäytön seuranta ja projektin raporttien kirjoittaminen pois-lukien asennus- ja käyttöohje. Suunnittelu-työtehtäviin kuuluvat sovelluksen, sen käyttöliittymän ja rajapintojen suunnittelu. Toteutus-työtehtäviin kuuluvat ohjelmointi, testaus, refaktorointi ja katselmoinnit. Vaatimusmäärittely-työtehtäviin kuuluvat vaatimusmäärittelyn ja käyttötapauksien kirjoitus.

Tässä luvussa käytettävät kaaviot on luotu 11.12.2006 ja kuvastavat sen ajankohdan ajankäyttöä. Kaavioiden suhdeluvut eivät projektin loppua kohti tule muuttumaan huomattavasti. Ainoastaan projektinhallinta-vaiheen tunnit kasvavat.

5.1 Ryhmän työnjako

Kuvassa 5.1 näkyy projektiryhmän jäsenten työtehtävien jakautuminen vaiheittain.



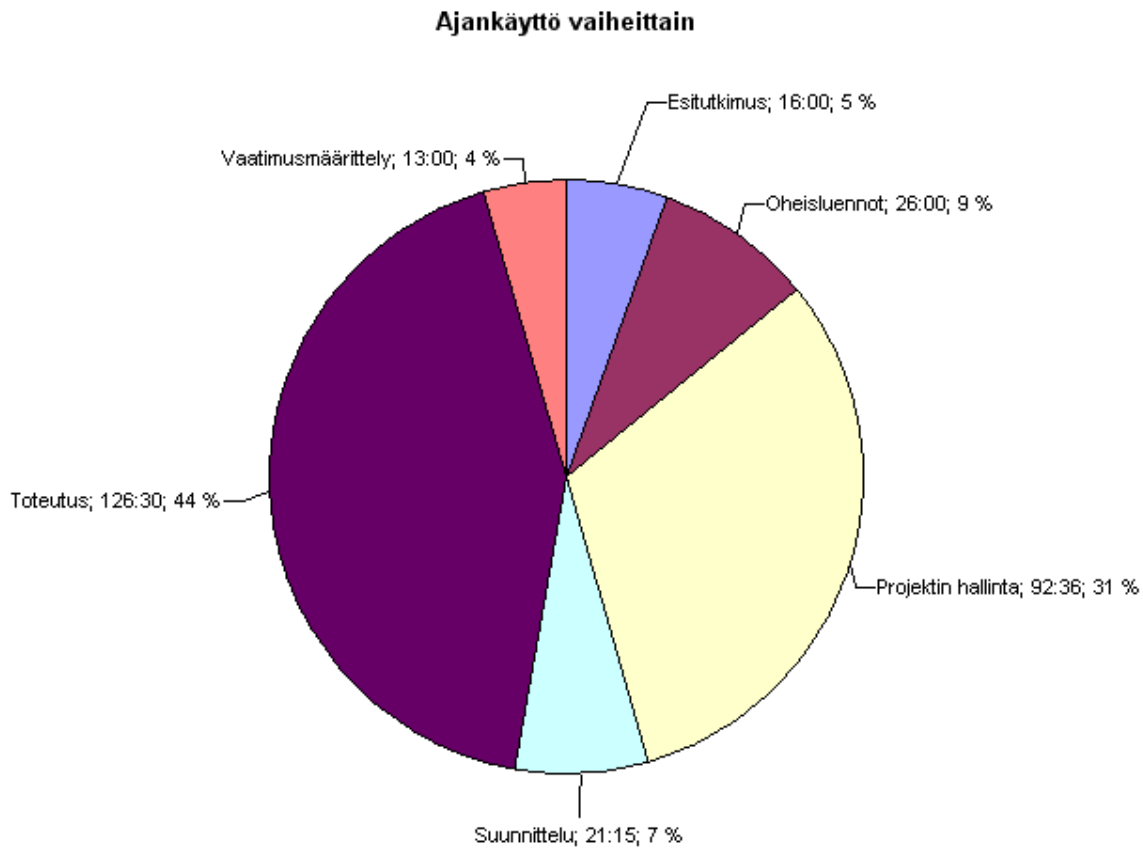
Kuva 5.1: Projektiryhmän ajankäyttö vaiheittain.

Työtehtävistä isoimmat kokonaisuudet muodostavat projektin hallinta ja toteutus. Projektin hallinnan koon suuruuteen osaltaan vaikuttaa se, että se sisältää useita osakokonaisuuksia kuten palaverit ja niihin liittyvät toiminnot, sekä tämän ja dokumentin [1] kirjoittamisen. Toinen syy sen kokoon liittyy käytettävään prosessimalliin. Prosessimallissa jokainen iteraatio suunnitellaan erikseen, joten projektipäällikön tuli hoitaa projektin suunnittelua läpi projektin. Asiakaslähtöisen prosessimallin takia kommunikointia asiakkaan kanssa tapahtui paljon sähköpostitse ja palavereissa.

Toteutuksesta tuli toiseksi isoin kokonaisuus. Toteutuksessa tehtiin paljon spikejä, eli toteutettavuustestejä. Lisäksi toteutettaessa tuli sovellusta myös suunniteltua, kun vanhat ratkaisut todettiin epäkelvoiksi. Toteutus sisälsi myös testauksen.

5.1.1 Ilari Liukko

Ilari Liukon työtehtävien jakautuminen näkyy kuvassa 5.2.

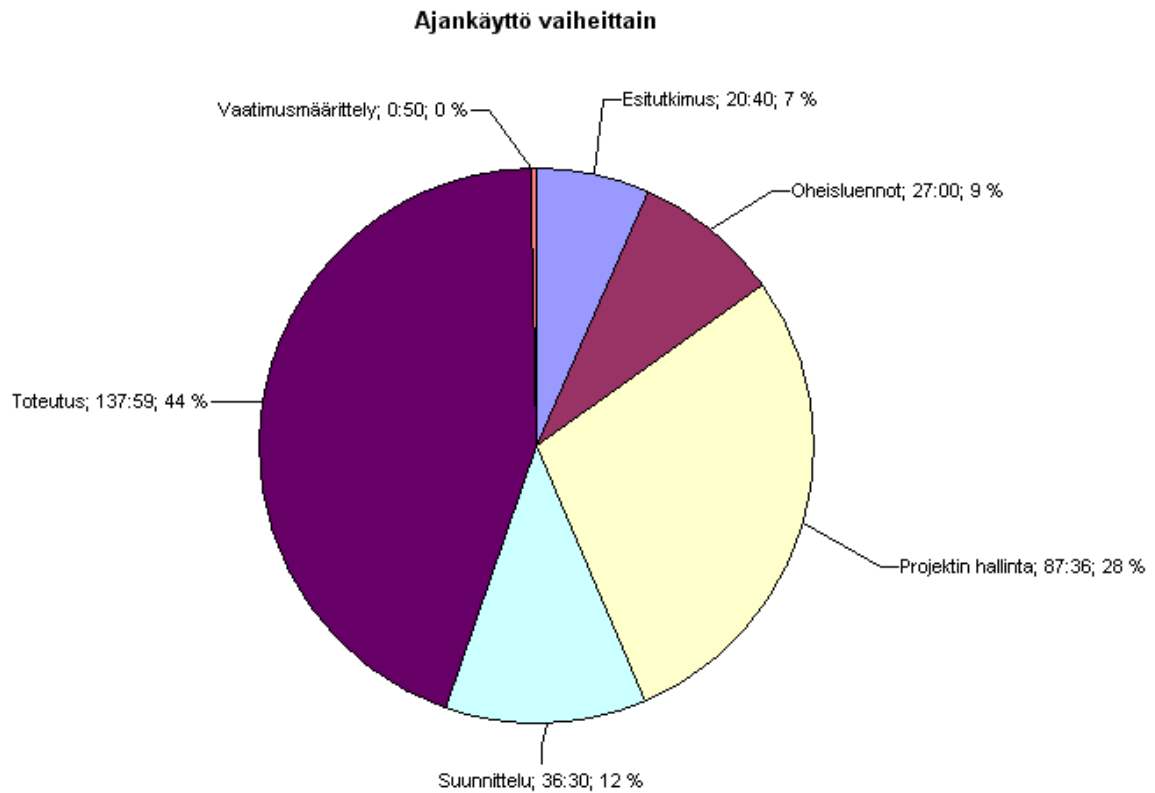


Kuva 5.2: Ilari Liukon ajankäyttö työvaiheittain.

Ilari Liukko halusi projektin alussa parantaa ohjelmointitaitojaan, tämän takia hänelle annettiin paljon ohjelmointitehtäviä. Hänen vastuullaan oli paljon käyttöliittymän eri osakokonaisuuksia.

5.1.2 Tuomo Pieniluoma

Tuomo Pieniluoma työtehtävien jakautuminen näkyy kuvassa 5.3.

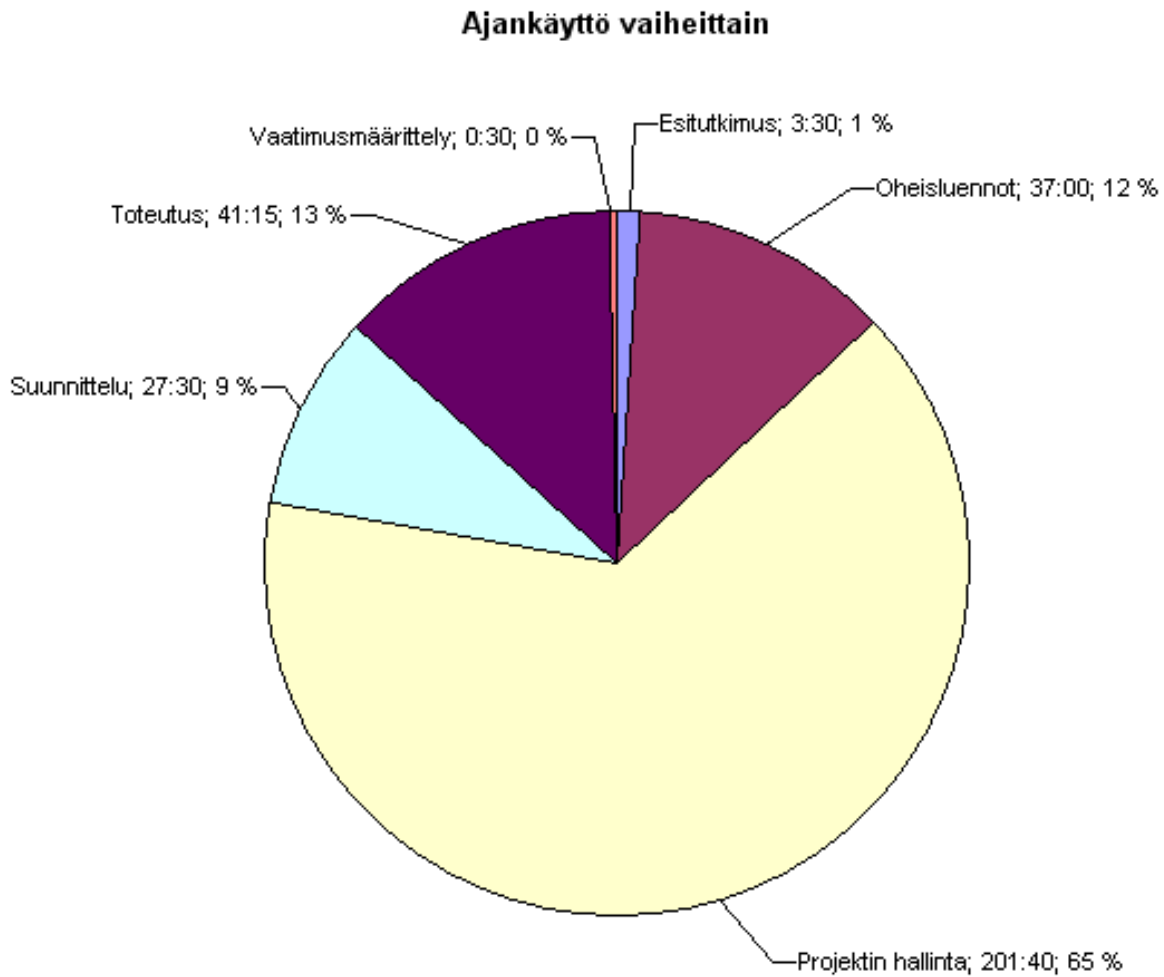


Kuva 5.3: Tuomo Pieniluoman ajankäyttö työvaiheittain.

Tuomo Pieniluoma toimi projektin päätoimisena tutkijana ja sekatyömiehenä. Tuomo toteutti Ilarin kanssa sovelluksen käyttöliittymän.

5.1.3 Vesa Pikki

Vesa Pikin työtehtävien jakautuminen näkyy kuvassa 5.4.

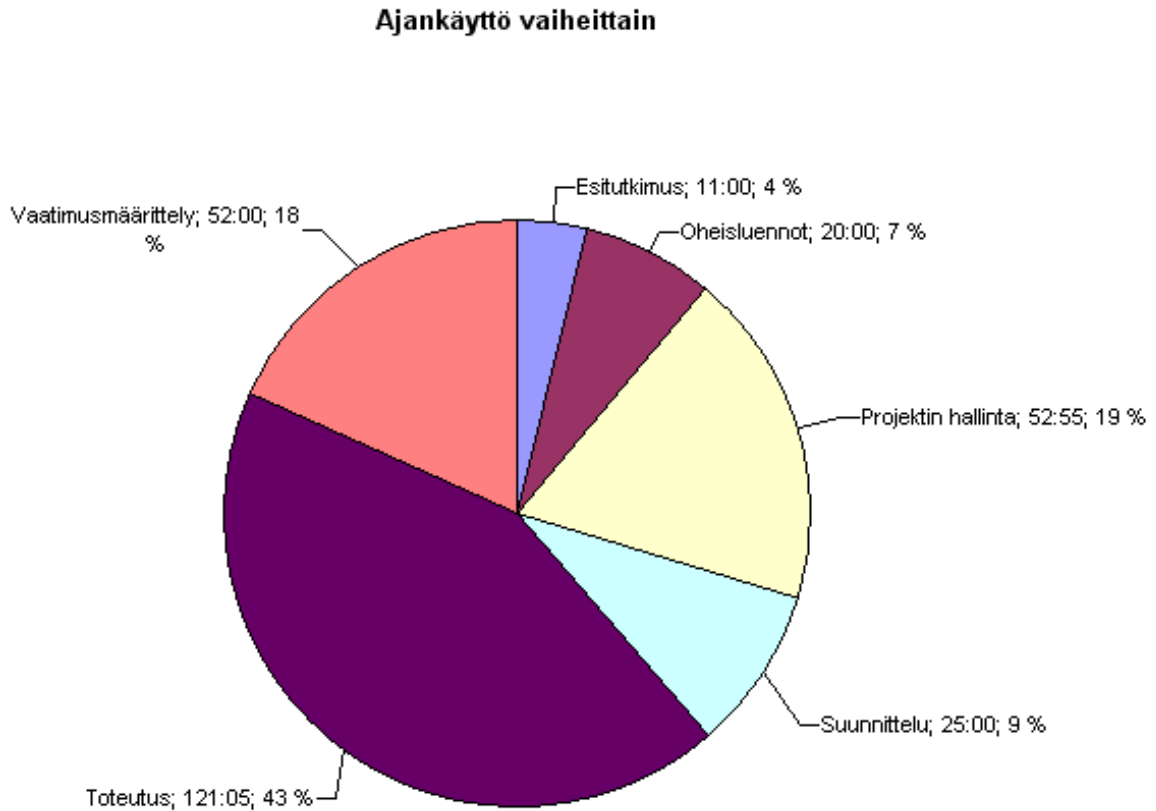


Kuva 5.4: Vesa Pikin ajankäyttö työvaiheittain.

Vesa Pikki toimi projektipäällikkönä, hänen vastuullaan oli projektin hallinta ja asiakasrajapinta. Vesa toimi Panun ohella sovelluksen suunnittelijana.

5.1.4 Panu Suominen

Panu Suomisen työtehtävien jakautuminen näkyy kuvassa 5.5.



Kuva 5.5: Panu Suomisen ajankäyttö työvaiheittain.

Panu Suomisen tehtäviin kuului sovelluksen suunnittelu, vaatimusmäärittely ja käytötapauksiin liittyvät ohjelmointitehtävät. Panu toteutti heuristiikan, jäsentimen ja käsitemallin.

6 Prosessimalli ja aikataulu

Tässä luvussa kerrotaan projektin käyttämästä prosessimallista, sekä sen aikataulun toteutumisesta. Aikatauluihin liittyvät kaaviot perustuvat viikkoon 49 asti kirjatuihin työtunneista.

6.1 Prosessimalli

Projekti toteutettiin käyttäen ketterää prosessimallia lyhyillä, noin 2-3 viikon iteraatioilla. Prosessimallille keskeistä on asiakaslähtöisyys. Projektiryhmällä oli muutamaa poikkeusta lukuunottamatta palaveri asiakkaan edustajan kanssa viikottain. Tämä mahdollisti nopean reagoinnin muutostilanteisiin ja mahdollisissa väärinymmärryksissä virheet korjattiin viimeistään viikon päästä. Prosessimalliin kuului myös spikejen, eli pikaisten testien ohjelmointia. Ne auttoivat ymmärtämään erilaisia pieniä toteutuskokonaisuuksia ja niillä pystyi helposti toteamaan, onko testattava toteutustapa mielekäs.

Jokaisen iteraation alussa projektiryhmä ehdotti asiakkaalle seuraavan iteraation tehtäviä, joista asiakas keskustelun kautta valitsi mielekkäimmät ja antoi niille prioriteetit. Iteraation välissä olevissa tarkistuspisteissä näistä vaatimuksista pystyttiin karsimaan pois ne, jotka eivät olleet mielekkäitä toteuttaa. Näitä havaittiin muun muassa spikejen avulla. Tarkistuspisteissa myös tarkennettiin epäselväksi jääneitä vaatimuksia. Iteraation lopussa asiakas tarkisti iteraation vaatimusten toteutumisen. Tämän jälkeen keskusteltiin seuraavan iteraation tehtävistä.

6.2 Iteraatioiden ajankäyttö

Projektsuunnitelmassa [1] oleva alkuperäinen iteraatioiden ajankäytön suunnitelma näkyy taulukossa 6.1.

Iteraatio	Kesto	Tärkeimmät tehtävät
0	1 Viikko	Aiheeseen tutustuminen, arkkitehtuurisuunnittelun aloitus, esitutkimus
1	2 Viikkoa	Vaatimusmäärittely, projektisuunnitelma, sovelluksen "ketjun"-todennus
Siirtymävaihe	1 Viikko	Edellisen iteraation tulosten hyväksyminen, seuraavan iteraation suunnitelmien tarkentaminen
2	2 Viikkoa	Sovelluksen ytimen laajennos, käyttöliittymän prototyyppi, testausraportin aloitus, testaus
Siirtymävaihe	1 Viikko	Edellisen iteraation tulosten hyväksyminen, seuraavan iteraation suunnitelmien tarkentaminen
3	2 Viikkoa	Rajapintojen toteutus, toimiva käyttöliittymä
Siirtymävaihe	1 Viikko	Edellisen iteraation tulosten hyväksyminen, seuraavan iteraation suunnitelmien tarkentaminen
4	2 Viikkoa	Käyttöliittymän viimeistely, rajapintojen viimeistely, testaus, sovellus- ja projektiraportin aloitus
Siirtymävaihe	1 Viikko	Edellisen iteraation tulosten hyväksyminen, seuraavan iteraation suunnitelmien tarkentaminen
5	3 Viikkoa	Projektin tulosten viimeistely ja hyväksyminen

Taulukko 6.1: Iteraatioiden suunnitellut taitekohdat.

Projektisuunnitelmassa [1] kuvattua siirtymävaihetta ei päädytty toteuttamaan sellaisenaan, sillä projektiryhmä pystyi esittämään tarkistuspisteissä ehdotuksen tulevan iteraation vaatimuksista. Tällöin siirtymävaiheelle ei ollut tarvetta. Tämän sijaan päätettiin käyttää aikaisemmin mainittua tarkistuspistekäytäntöä. Tällöin jokaisen iteraation välissä pidettiin epävirallinen palaveri, jossa tarkennettiin vaatimuksia. Iteraatioita projektin aikana toteutettiin alkuperäisestä suunnitelmasta poiketen nollaiteraation lisäksi kuusi. Iteraatioiden ajankäytön suunnitelmat ja toteutumat löytyvät liitteistä A.1, A.2, A.3, A.4 ja A.5.

Projektiryhmä sopi keskenään aloittavansa kovalla työtahdilla projektin päästäkseen nopeasti perille projektityöskentelystä, estääkseen sen, ettei myöhäisessä vaiheessa tule kiire ja että projektin alkupään työt saadaan tehtyä ajallaan. Tämä koettiin erittäin hyväksi ratkaisuksi.

Projekti alkoi viikolla 37 niin sanotulla nollaiteraatiolla, jonka aikana projektiryhmä kävi keskustelua asiakkaan kanssa sovelluksen yleisistä vaatimuksista, valitsi työkalut ja sopi käytänteistä.

Tämän jälkeen alkoi viikolla 39 ensimmäinen iteraatio, jonka aikana projektiryhmä laati asiakkaan kanssa käyttötapauksia, kirjoitti projektisuunnitelmaa ja vaatimusmäärittelyä, sekä teki spiken Abbottin heuristiikan toteuttamisesta. Ensimmäisen iteraation aikana sovittiin myös Stanfordin jäsentimen käyttämisestä. Projektin hallinnan laajuus yllätti projektipäällikön lähes jokaiselta sen osa-alueelta. Tämän seurauksena aikaa käytettiin huomattavasti suunnittelua enemmän.

Toisen iteraation kohdalla lisättiin syöteformaatti ja toteutettiin käyttöliittymän prototyyppi. Toinen iteraatio alkoi viikolla 41. Ajankäyttöön oli määritelty liikaa aikaa sovelluksen suunnitteluun, sillä sovellusta oli kattavasti suunniteltu ennakkoon edellisessä iteraatiossa.

Kolmas iteraatio alkoi viikolla 43 ja oli projektin ainoa kolmiviikkoinen iteraatio. Kolmannen iteraation aikana päätehtäviä oli implementoida Stanfordin jäsentimen ja käsitelmän piirto dot-sovellusta käyttäen. Ohjelmointi tapahtui suunniteltua helpommin, mutta projektin hallintaan käytetty aika ylitti suunnitellun hieman palaverihin valmistelun osalta. Tähän vaikutti palaverien dokumenttien suuri määrä. Suunnitteluun käytettiin kyseisessä iteraatiossa suunniteltua vähemmän aikaa. Suunnittelua tapahtui paljon sovelluksen kehittämisen yhteydessä ja sitä oli tapahtunut paljon jo aikaisemmissa iteraatioissa.

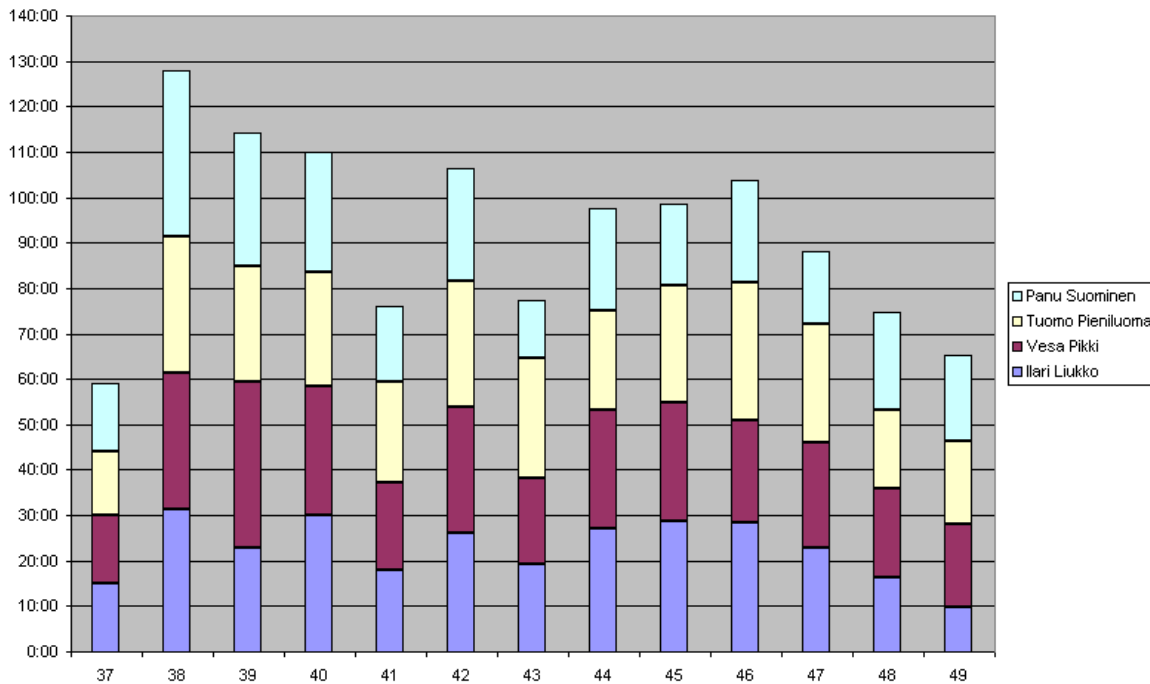
Neljäs iteraatio alkoi viikolla 46. Neljännessä iteraatiossa parannettiin sovelluksen

käyttöliittymää ja yleistä käytettävyyttä. Ajankäytön suunnitelma oli suhteellisen tarkka arvio käytetystä ajasta.

Viikolla 48 alkanut viides iteraatio oli viimeinen toteutusiteraatio. Siinä lisättiin käyttöliittymään toimintoja ja korjattiin sovelluksen virheitä. Viidennen iteraation kohdalle osui projektiryhmän jäsenille tenttejä sekä itsenäisyyspäivä. Näiden takia ryhmän jäsenillä ei ollut käytettävissä yhtä paljon aikaa kuin aikaisemmin. Vaatimuksista karsittiin tarkistuspisteessä osa ominaisuuksista pois, jonka takia toteutettavia ominaisuuksia ei ollut yhtä paljoa kuin alunperin on ajankäyttöön suunniteltu. Kaikki ominaisuudet projektiryhmän jäsenet saivat kuitenkin toteutettua ajoissa, sekä projektiryhmän jäsenet ehtivät aloittaa raporttien kirjoitusta.

6.3 Viikkotuntien jakautuminen

Ryhmän yhteinen viikkotuntien jakautuminen näkyy kaaviossa 6.1.

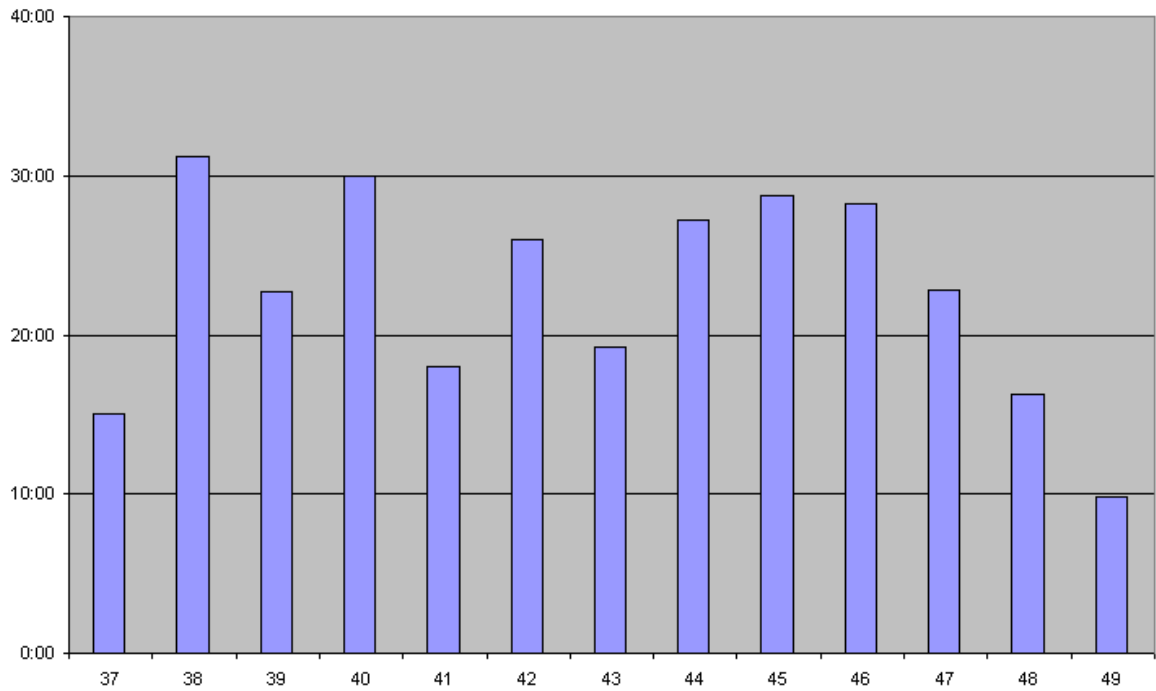


Kuva 6.1: Ryhmän viikkotunnit.

Projektiryhmä tietoisesti aloitti projektin intensiivisesti. Alussa viikkotunnit olivat korkeita, mutta laskivat hitaasti viikkoon 41 mennessä, kun projektisuunnitelma sekä vaatimusmäärittelyn runko saatiin valmiiksi. Sen jälkeen projektiryhmä aloitti toisen intensiivisen jakson, joka hitaasti laski viikkoon 49 asti. Tämän intensiivisen jakson aloitti toisen iteraation aikana toteutettu käyttöliittymän prototyyppi. Kolmas iteraatio oli tasaisesti työläs iteraatio, jonka aikana suurimmat toiminnallisuudet sovellukseen toteutettiin. Tämän jälkeen loppuprojekti koostui lähinnä pienten osakokonaisuuksien toteuttamisesta ja bugien korjauksesta, joten viikkotunnit vähenivät hitaasti.

6.3.1 Ilari Liukko

Ilarin työtunnit viikottain näkyvät kaaviossa 6.2.

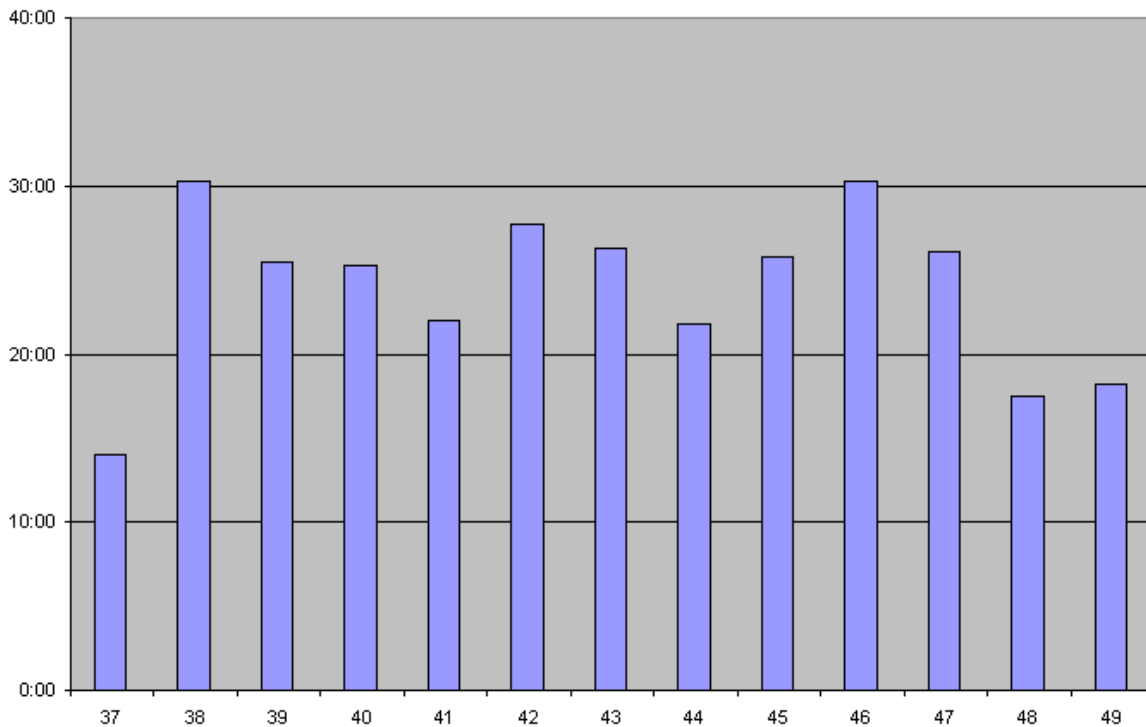


Kuva 6.2: Ilarin viikkotunnit.

Ilarin työtunnit (kuva 6.2) noudattavat suhteellisen tarkkaan projektiryhmän työtunteja. Viikolla 49 Ilari oli puolet viikosta pois tenttien takia.

6.3.2 Tuomo Pieniluoma

Tuomon työtunnit viikottain näkyvät kaaviossa 6.3.

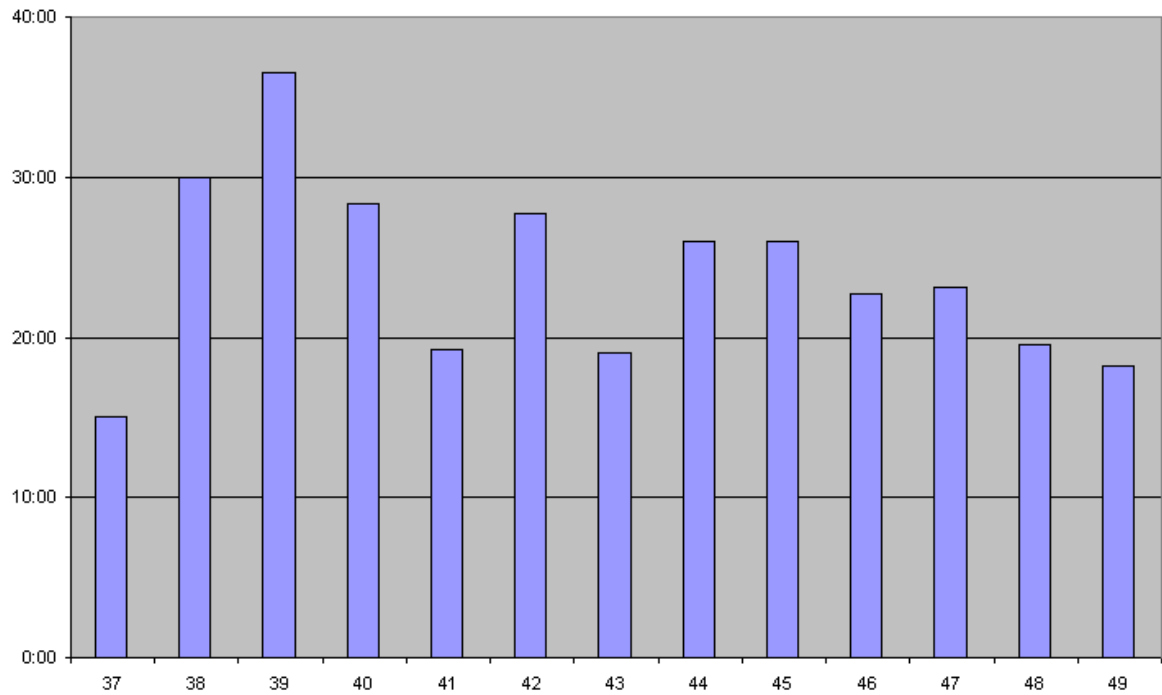


Kuva 6.3: Tuomon viikkotunnit.

Tuomon työtunnit (kuva 6.3) ovat melko tasaisia läpi projektin. Projektin loppua kohti Tuomon viikkotunnit kasvoivat selvästi muihin verrattuna. Tuomolle tuli Vesan kanssa eniten työtunteja projektiryhmästä.

6.3.3 Vesa Pikki

Vesan työtunnit viikottain näkyvät kaaviossa 6.4.

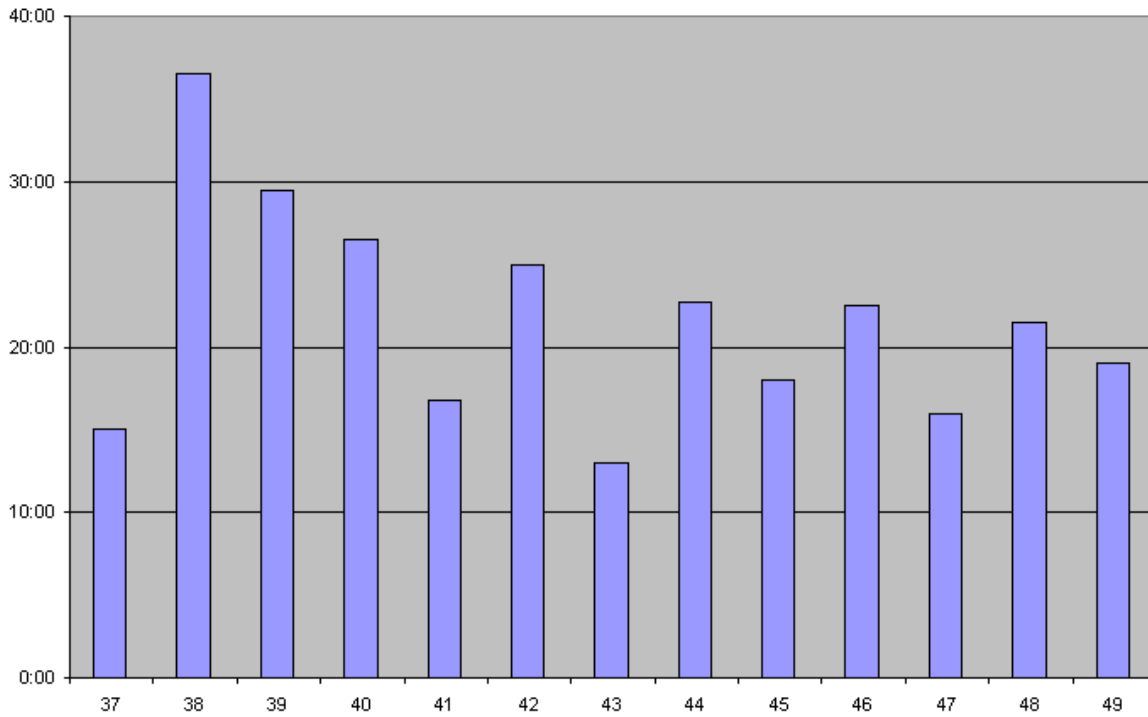


Kuva 6.4: Vesan viikkotunnit.

Vesan työtunnit (kuva 6.4) ovat tasaisia, sillä hänellä oli säännölliset työajat. Vesalle tuli Tuomon ohella eniten työtunteja projektiryhmästä.

6.3.4 Panu Suominen

Panun työtunnit viikottain näkyvät kaaviossa 6.5.



Kuva 6.5: Panun viikkotunnit.

Panun työtunnit 6.5 alkoivat erittäin intensiivisesti, mutta hiipuivat loppua kohti. Kunnes Panu alkoi viikolla 44 kirimään muuta projektiryhmää työtunneissa kiinni.

7 Riskit ja niiden seuranta

Tässä luvussa kuvataan projektin läpivientiin liittyviä riskejä, mitkä niistä tapahtuivat ja onnistuttiinko niihin reagoimaan oikein.

Taulukossa *S. vaikutus* tarkoittaa suunniteltua vaikutusta, sekä *T. vaikutus* toteutunutta vaikutusta. Riskit on listattu taulukossa 7.1 käyttäen pieni / kohtalainen / suuri -asteikkoa todennäköisyydelle, vaikutukselle ja toteumalle.

Riski	Todennäk.	S. vaikutus	T. vaikutus	Toteutui
Henkilöihin liittyvät riskit				
Projektiryhmän jäsenten poissaolot	Pieni	Kohtalainen	Pieni	Kyllä
Ohjaajien poissaolo	Pieni	Kohtalainen	Pieni	Kyllä
Tilaaajan edustajien poissaolo	Pieni	Suuri	-	Ei
Kokemuksen puute	Suuri	Kohtalainen	Pieni	Kyllä
Viestinnän ongelmat	Kohtalainen	Suuri	Pieni	Kyllä
Toteutukseen liittyvät riskit				
Laitteisto- ja ohjelmisto-ongelmat	Pieni	Suuri	Pieni	Kyllä
Ongelmat ohjelmoinnissa	Pieni	Kohtalainen	-	Ei
Aikataulun suunnittelu	Kohtalainen	Suuri	-	Ei
Vaativuuden muuttuminen	Kohtalainen	Kohtalainen	Pieni	Kyllä
Sovellusaluekohtaiset riskit				
Jäsentimen valinta	Kohtalainen	Suuri	Kohtalainen	Kyllä
Heuristiikan soveltaminen	Kohtalainen	Suuri	-	-
Käytettävät tallenusmuodot	Kohtalainen	Kohtalainen	Pieni	Kyllä

Taulukko 7.1: Projektin läpivientiin liittyviä riskejä

Projektiryhmä onnistui hyvin ennakoimaan projektin etenemistä ja lopputulosta mahdollisesti haittaavat riskit. Onnistuneella työtehtävien ja vastuiden hajauttamisella varauduttiin henkilöihin liittyviin riskeihin ja tämä toimi projektiryhmällä erinomaisesti. Toteutukseen liittyvät riskit minimoitiin hyvin käytetyllä prosessimallilla. Sovellusaluekohtaisten riskien vaikutus saatiin minimoitua kattavalla esitutkimuksella ja teknisen tuen hyödyntämisellä.

7.1 Henkilöihin liittyvät riskit

Projektiryhmän jäsenten poissaoloja oli projektin aikana varsin vähän. Jäsenille ker-tyi joitakin yksittäisiä poissaoloja projektin viikkopalavereista. Tämän lisäksi oli muutamia yksittäisiä työpäiviä, jolloin joku projektiryhmän jäsenistä ei voinut osal-listua projektin tekemiseen. Kaikkiaan projektiryhmän jäsenten vähäisten poissao-lojen vaikutus projektin etenemiseen ja lopputulokseen oli pieni. Vaikutus onnis-tuttiin minimoimaan hyvin vastuiden hajauttamisella ja työtehtävien jakamisella useammalle ryhmän jäsenelle.

Projektin kuluessa vastaava ohjaaja oli poissa yhdestä viikkopalaverista ja tämän ta-kia kyseinen viikkopalaveri ei ollut päätösvaltainen. Päätösvaltaisuuden puuttuessa tulevista tehtävistä sovittiin alustavasti ja muodolliset päätökset toimitettiin vastaa-valle ohjaajalle hyväksyttäväksi sähköpostitse. Tekninen ohjaaja oli poissa yhdestä viikkopalaverista, sillä kuitenkin ei ollut vaikutusta päätösvaltaisuuteen. Yleisesti ohjaajien poissaolojen vaikutus projektin etenemiselle ja lopputulokselle oli hyvin pieni.

Ainakin toinen tilaajien edustaja oli paikalla kaikissa projektin viikkopalavereissa ja tarkistuspisteissä, joten riski heidän poissaoloistaan ei toteutunut.

Projektiryhmän jäsenten kokemus ja oppimiskyky tukivat saumatonta projektin tek-nistä toteuttamista. Projektin alkuvaiheessa kokemuksen puutteen vuoksi työmää-rät asiakirjojen suhteen arvioitiin kuitenkin liian pieniksi ja tällä oli pieni vaikutus projektin etenemiselle projektin alussa. Tilanne kuitenkin tasottui nopeasti, kun pro-jektin alun asiakirjat saatiin valmiiksi.

Projektin alkuvaiheessa esiintyi eriävyyksiä termistöstä projektiorganisaation jäsen-ten kesken ja tämä osaltaan hidasti projektin käynnistämistä jonkin verran, mutta termistön perusteellisen läpikäynnin jälkeen tätä ongelmaa ei enää esiintynyt.

7.2 Toteutukseen liittyvät riskit

Projektin alussa oli pieniä ohjelmisto-ongelmia, joiden selvittämiseen ATK-tuella meni oma aikansa. Lopullinen vaikutus projektin etenemiselle ja lopputulokselle laitteisto- ja ohjelmisto-ongelmilla oli kuitenkin vähäinen.

Varsinaisia ongelmia ohjelmoinnissa ei ilmennyt. Muutaman vaatimuksen toteu-tuminen jäi kuitenkin iteraation päättyessä varmistamatta ohjelmassa ilmenneiden

virheiden takia, mutta kyseiset virheet korjattiin aina seuraavana työpäivänä.

Aikataulujen suunnittelussa ilmenneet pienet arviointivirheet eivät vaikuttaneet projekti etenemiseen tai lopputulokseen.

Vaatimusten muuttuminen kesken projektin vaikutti hiukan projektin etenemiseen, johtuen uusien ja muuttuneiden vaatimusten aiheuttamasta työmäärästä. Käytetyn prosessimallin ansiosta vaatimusten muuttumisen vaikutus projektin lopputulokseen saatiin hyvin minimoitua ja kyseisiin tilanteisiin pystyttiin vastaamaan nopeasti.

7.3 Sovellusaluekohtaiset riskit

Jäsentimen valinnan pitkittyminen vaikutti osaltaan sovelluksen kehitykseen ja sitä kautta projektin etenemiseen. Oikean jäsentimen etsimiseen kului projektin alussa kohtuullisen runsaasti aikaa. Ilmaisen suomenkielisen jäsentimen puuttuessa päädyttiin englanninkieliseen jäsentimeen, ja projektin alussa kehitetty testimateriaali jäi sellaisenaan hyödyttömäksi.

Heuristiikan soveltamisessa ei ilmennyt projektin kannalta mitään ongelmia. Käytetty heuristiikka oli hyvin suoraviivainen ja helppo toteuttaa.

Käytettävien tallennusmuotojen kanssa ilmeni joitakin ongelmia, jotka aiheuttivat projektiryhmän jäsenille lisää työtä. Ongelmat ratkaistiin kuitenkin ripeästi ja projektin kannalta epäoleelliset tallennusmuodot voitiin käytetyn prosessimallin ansiosta nopeasti vaihtaa tai jättää huomiotta.

8 Kokemuksia projektista

Jokainen ryhmän jäsen kertoo kokemuksiaan projektista, sekä mitä projektin aikana oppi.

8.1 Ilari Liukko

Projekti oli mielenkiintoinen opintojakso, jossa pääsi oikeasti soveltamaan kursseilla opittuja asioita. Käytetyt työkalut olivat suurimmaksi osaksi tuttuja, mutta ohjelmoinnissa tunnen kehittyneeni. Varsinkin Javan 1.5 version uudet ominaisuudet olivat suurimmaksi osaksi ennestään tuntemattomia. Myös Swing-kirjaston komponenttien muokkaaminen oli ennen projektia hieman hatarala pohjalla.

Kokemusta sain lisää myös raporttien ja muiden papereiden kirjoittamisesta, palaverikäytännöistä ja yleisölle esiintymisestä.

8.2 Tuomo Pieniluoma

Ennen projektin alkua luulin projektin olevan enemmän tekniseen toteutukseen painottuva opintokokonaisuus, mutta sen odotettua suurempi sosiaalinen painotus oli lopulta positiivinen yllätys. Projektin teknisen toteutuksen jäätyä vaatimustasoltaan sen verran alhaiseksi, että en oppinut uusia ohjelmointiin tai muuhun ohjelmiston tekniseen toteutukseen liittyviä taitoja, mutta vanhoja taitoja tuli kehitettyä jonkin verran. Yksittäisenä ennestään hiukan tuntemattomampana teknisenä ratkaisuna muistuu mieleen sovelluksen suorittaminen useissa säikeissä, jonka toteutus osui minun vastuulleni.

Suurin osa projektin oppimiskokemuksistani kuitenkin tulee itse prosessimallista, projektityöskentelystä, palaverikäytännöistä ja muusta sosiaalisesta kanssakäymisestä. Sovellusalueen vuoksi myös kielio pillistä terminologiaa tuli kerrattua.

8.3 Vesa Pikki

Minulla oli työkokemusta ohjelmoinnista projektia ennen, joten ilmaisin sovellusprojektin ilmottautumislomakkeessa halukkuuteni toimia projektipäällikkönä. Päätelin oppivani enemmän ohjelmistokehityksestä sitä kautta. Projektin alkaessa muu projektiryhmä mielellään luovutti tämän työläältä kuulostavan työn minulle. Projektipäällikön tehtävissä opinkin paljon ohjelmistoprojektien hallinnasta ja ajankäytön suunnittelusta. Samoin projektipäällikön tehtävissä prosessimalli tuli erittäin tutuksi. Sain myös kokemusta kirjoittamisesta ja parannettua taitojani sen osalta. Palaverikäytännöt olivat myös uusi asia minulle. Ohjelmointia en paljoa projektin aikana harrastanut, eikä projektin aikana tullut mitään uutta sen saralta opittua, paitsi Panulta muutamia Javan kikkoja, jotka lähinnä liittyivät uuden Javan version ominaisuuksiin.

Tunnelma mielestäni koko projektin läpi oli erittäin hyvä ja projektitilassa on monena päivänä vedet silmissä naurettu. Hyvässä seurassa aika kului todella nopeasti.

8.4 Panu Suominen

Koska projektin aikana käytetyt työkalut olivat ennestään tuttuja en kokenut projektin aikana oppivani mitään konkreettisia uusia taitoja. Suurimat oppimiskokemukseni liittyvät ihmisten käytökseen ja vuorovaikutukseen toistensa kanssa. Lisäksi havaitsin, että saatan käytökselläni antaa hiukan väärän kuvan itsestäni, mikä voi vaikuttaa tuntemattomampien kanssa toimimiseen negatiivisella tavalla.

Ohjelmointityöstä havaitsin, että hyvä moduulijako auttaa huomattavasti töiden jakamista ja tämä onnistuikin projektin aikana hyvin. Kuitenkin koodia pitää tutkia ristiin enemmän, jotta kaikki pysyvät helpommin samalla viivalla koodin ymmärtämisen kanssa ja koodi näyttää kaikkialla samalla.

Projekti oli kaikenkaikkiaan mukavaa aikaa ja muun ryhmän kanssa kommunikointi ja työskentely sujui erinomaisesti.

9 Lähteet

- [1] Ilari Liukko, Tuomo Pieniluoma, Vesa Pikki ja Panu suominen, "UCOT-Sovellusprojekti Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 2006.
- [2] Ilari Liukko, Tuomo Pieniluoma, Vesa Pikki ja Panu suominen, "UCOT-Sovellusprojekti Sovellusraportti", Jyväskylän yliopisto, tietotekniikan laitos, 2006.
- [3] Ilari Liukko, Tuomo Pieniluoma, Vesa Pikki ja Panu suominen, "UCOT-Sovellusprojekti Vaatimusmäärittely", Jyväskylän yliopisto, tietotekniikan laitos, 2006.
- [4] Ilari Liukko, Tuomo Pieniluoma, Vesa Pikki ja Panu suominen, "UCOT-Sovellusprojekti Testausraportti", Jyväskylän yliopisto, tietotekniikan laitos, 2006.

A Iteraatioiden ajankäytön suunnitelmat

Projektin iteraatioiden suunnitellut ajankäytöt ja niiden toteutumat.

A.1 Ensimmäinen iteraatio

Tehtävä	Ilari		Tuomo		Vesa		Panu		Yhteensä	
	Suunn.	Tot.	Suunn.	Tot.	Suunn.	Tot.	Suunn.	Tot.	Suunn.	Tot.
Projektin hallinta	7	19	27	37	39	58,5	7	14,5	80	129
- seuranta ja tiedotus	0	1	2	0	11	17	0	0	13	18
- ajankäytön hallinta	0	1,75	2	2	2	2,25	0	0	4	6
- projektin suunnittelu (projektsuunnitelma)	0	3	15	19,25	15	28,75	0	1	30	52
- viikkopalaveri	4	6	4	6	4	6	4	6	16	24
- valmistelu	0	2	2	7,25	4	4,5	0	2,75	6	16,5
- pöytäkirja	3	5,25	0	1	3	0	3	4,75	9	11
- www-sivujen toteutus	0	0	2	1,5	0	0	0	0	2	1,5
Esitutkimus	0	0	8	4,25	0	0	0	0	8	4,25
- jäsentimeen perehtyminen	0	0	8	4,25	0	0	0	0	8	4,25
Vaatimusmäärittely	27	11	0	0,75	0	0	27	28,5	54	40,25
- kirjoitus	10	2,5	0	0,75	0	0	10	13,5	20	16,75
- käytötapauskuvausten esitutkimus	5	2	0	0	0	0	5	6	10	8
- käytötapauskuvausten palaveri	2	2	0	0	0	0	2	2	4	4
- käytötapauskuvausten kirjoitus	10	4,5	0	0	0	0	10	7	20	11,5
Sovelluksen suunnittelu	0	0	2	2	0	1	2	1,5	4	4,5
- arkkitehtuurisuunnitelma	0	0	0	2	0	1	2	1,5	2	4,5
- rajapintojen suunnittelu	0	0	2	0	0	0	0	0	2	0
Toteutus	7	8	4	0	0	0	5	12,5	16	20,5
- toteutuksen osavaiheet	7	8	4	0	0	0	5	12,5	16	20,5
- testaus	0	0	0	0	0	0	0	0	0	0
Oheisluennot	4	3	4	3	6	5	4	1	18	12
- Bugzilla perehdytys	2	1	2	1	2	1	2	1	8	4
- CVS-luento	2	2	2	2	2	2	2	0	8	6
- projektipäälliköiden tapaaminen	0	0	0	0	2	2	0	0	2	2
Tunteja yhteensä	45	41	45	47	45	64,5	45	58	180	210,5

Kuva A.1: Ensimmäisen iteraation ajankäytön suunnitelma.

A.2 Toinen iteraatio

Tehtävä	Ilari		Tuomo		Vesa		Panu		Yhteensä	
	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.
Projektin hallinta	11	8,25	7	4,25	27	21,25	6	4,25	51	38
- seuranta ja tiedotus	0	0	0	0	10	10,75	0	0	10	10,75
- ajankäytön hallinta	1	0	1	1	4	4,75	1	0	7	5,75
- projektin suunnittelu	0	0,5	0	1	4	2,5	0	0	4	4
- viikkopalaveri	4	1,25	4	1,25	4	1,25	4	1,25	16	5
- valmistelu	1	1	2	1	5	2	1	3	9	7
- pöytäkirja	5	5,5	0	0	0	0	0	0	5	5,5
Esitutkimus	0	5,5	4	2,75	0	0	0	0	4	8,25
- jäsentimeen perehtyminen	0	0	4	2,75	0	0	0	0	4	2,75
- Grappaan perehtyminen	0	5,5	0	0	0	0	0	0	0	5,5
Vaatusmäärittely	1	1	0	0	0	0	10	7,5	11	8,5
- kirjoitus	1	1	0	0	0	0	10	7,5	11	8,5
Sovelluksen suunnittelu	14	4,5	27	18,5	15	14,5	20	7,5	76	45
- arkkitehtuurisuunnitelma	5	0	2	2	3	5,5	10	5,5	20	13
- rajapintojen suunnittelu	2	0	10	0,5	3	7	5	0	20	7,5
- käyttöliittymän suunnittelu	7	4,5	5	6	4	1,5	5	2	21	14
- testausraportti	0	0	10	10	5	0,5	0	0	15	10,5
Toteutus	22	17,75	10	14,25	6	3	12	14	50	49
- käyttötapauskuvauksen luku tiedostosta	5	0	0	0	2	1	5	4,75	12	5,75
- käyttöliittymän prototyyppi ja integrointi	15	16,75	5	10,75	2	2	5	8,25	27	37,75
- testaus	2	1	5	3,5	2	0	2	1	11	5,5
Oheisluennot	7	7	7	7	7	7	7	7	28	28
- käytettävyysluento	7	7	7	7	7	7	7	7	28	28
Iunteja yhteensä	55	44	55	46,75	55	45,75	55	40,25	220	176,75

Kuva A.2: Toisen iteraation ajankäytön suunnitelma.

A.3 Kolmas iteraatio

Tehtävä	Ilari		Tuomo		Vesa		Panu		Yhteensä	
	Suur	Tot.	Suur	Tot.	Suur	Tot.	Suur	Tot.	Suun.	Tot.
Projektin hallinta	6	11,25	9	14,25	27	34,5	9	9,25	51	69,25
- seuranta ja tiedotus	0	0	0	0,5	15	15,25	0	0	15	15,75
- ajankäytön hallinta	0	0	0	0,75	4	4	0	0	4	4,75
- viikkopalaveri	4	5,25	4	5,25	4	3,25	4	5,25	16	19
- valmistelu	2	6	0	3,25	4	12	0	1	6	22,25
- pöytäkirja	0	0	5	4,5	0	0	5	2	10	6,5
- projektikansion päivitys	0	0	0	0	0	0	0	1	0	1
Esitutkimus	10	6,5	0	2,5	2	1	5	5	17	15
- jäsentimeen perehtyminen	0	0	0	2,5	1	0	5	2	6	4,5
- dotiin perehtyminen	10	6,5	0	0	1	0	0	0	11	6,5
- GXL:n perehtyminen	0	0	0	0	0	1	0	3	0	4
Vaatusmäärittely	0	0	0	0	0	0	2	2	2	2
- kirjoitus	0	0	0	0	0	0	2	2	2	2
Sovelluksen suunnittelu	6	3	11	5,25	7	3,5	6	3	30	14,75
- arkkitehtuurisuunnitelma	2	0	2	0	2	2	2	1	8	3
- rajapintojen suunnittelu	2	0	2	0	2	0	2	0	8	0
- käyttöliittymän suunnittelu	2	3	2	1,75	2	1,5	2	2	8	8,25
- testausraportti	0	0	5	3,5	1	0	0	0	6	3,5
Toteutus	45	43,5	47	42,25	27	18,75	45	25,5	164	130
- Stanfordin jäsentimen integrointi	0	0	5	2,5	5	0	15	12,5	25	15
- käyttöliittymän muutokset	21	32,75	20	14,25	5	0,75	5	0	51	47,75
- Analyysimallin piirtäminen	22	4,75	7	4,75	5	0	0	0	34	9,5
- processiml input adapterin toteutus	0	0	0	0	5	8,5	12	0	17	8,5
- käsitemallin muokkaus	0	0	0	0	0	4	9	5	9	9
- rajapintojen toteutus	0	0	10	0	5	0	2	1	17	1
- testaus	2	1,5	5	8,5	2	2,5	2	0	11	12,5
- lähdekoodin refaktorointi	0	4,5	0	12,25	0	3	0	7	0	26,75
Väliesittely	5	3	5	4	9	10	5	0	24	17
- valmistelu	2	1	2	1	5	6,5	2	0	11	8,5
- esittäminen	2	2	2	2	2	2	2	0	8	6
- raportin kirjoitus	1	0	1	1	2	1,5	1	0	5	2,5
Oheisluennot	3	3	3	3	3	3	3	3	12	12
- tekijänoikeusluento	3	3	3	3	3	3	3	3	12	12
Tunteja yhteensä	75	70,25	75	71,25	75	70,75	75	47,75	300	260

Kuva A.3: Kolmannen iteraation ajankäytön suunnitelma.

A.4 Neljäs iteraatio

Tehtävä	Ilari		Tuomo		Vesa		Panu		Yhteensä	
	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.
Projektin hallinta	9	7,5	6	3,75	34	23	6	3,75	55	38
- seuranta ja tiedotus	0	0	0	0	10	12	0	0	10	12
- ajankäytön hallinta	0	0	0	0	3	2,25	0	0	3	2,25
- viikkopalaveri	4	3,75	4	3,75	4	3,75	4	3,75	16	15
- valmistelu	2	0	2	0	10	2	2	0	16	2
- pöytäkirja	3	3,75	0	0	3	0	0	0	6	3,75
- sovellusraportin valmistelu	0	0	0	0	4	3	0	0	4	3
Esitutkimus	0	0	0	0	3	2,5	3	0	6	2,5
- GXL:n perehtyminen	0	0	0	0	3	2,5	3	0	6	2,5
Vaatimusmäärittely	0	0	0	0	0	0	2	1,5	2	1,5
- kirjoitus	0	0	0	0	0	0	2	1,5	2	1,5
Sovelluksen suunnittelu	2	0	2	0	2	0	2	0	8	0
- arkkitehtuurisuunnitelma	1	0	1	0	1	0	1	0	4	0
- käyttöliittymän suunnittelu	1	0	1	0	1	0	1	0	4	0
Toteutus	44	41,75	47	46,75	16	12,5	42	36,5	149	137,5
- Käyttöliittymän muokkaus	15	31,75	20	23	0	0,25	5	1	40	56
- Asetus-dialogi	10	0	5	4	0	2	0	0	15	6
- Käsitemallin tallennus GXL-muodossa	0	0	0	0	5	0	10	2,5	15	2,5
- Käsitemallin lataus ja tallennus	0	0	5	0	0	0	5	0	10	0
- Käsitemallin muokkaukset	0	0	0	0	0	0	15	12,5	15	12,5
- Käsitemallin kuvan tallennus	3	0	0	0	0	3,5	0	0	3	3,5
- Muokkausloki	10	0	5	0	5	0	0	1	20	1
- Jäsennys ja heuristiikka säikeillä ajettavaksi	4	0	5	9,75	2	0	5	2,5	16	12,25
- Linux yhteensopivuus	0	0	2	0	2	3	0	0	4	3
- Testaus	2	0	5	0	2	0	2	0	11	0
- Ilmenneiden bugien korjaus	0	3	0	4	0	0	0	5,5	0	12,5
- Lähdekoodin refaktorointi	0	5	0	4	0	1,75	0	9,5	0	20,25
- Katselmointi	0	2	0	2	0	2	0	2	0	8
Tunteja yhteensä	55	49,25	55	50,5	55	38	55	41,75	220	179,5

Kuva A.4: Neljännen iteraation ajankäytön suunnitelma.

A.5 Viides iteraatio

Tehtävä	Ilari		Tuomo		Vesa		Panu		Yhteensä	
	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.	Suun.	Tot.
Projektin hallinta	9	13,25	8	4,75	31	24,5	5	1,75	53	44,25
- Seuranta ja tiedotus	0	0	0	0	10	7,5	0	0	10	7,5
- Ajankäytön hallinta	0	0	0	0	3	3	0	0	3	3
- Viikkopalaveri	4	4,5	4	4,5	4	4	4	1,75	16	14,75
- Valmistelu	0	0,5	0	0,25	5	2,5	1	0	6	3,25
- Pöytäkirja	0	0	4	0	4	4,5	0	0	8	4,5
- Sovellusraportti	5	8,25	0	0	0	0	0	0	5	8,25
- Projektiraportti	0	0	0	0	5	3	0	0	5	3
Vaativuusmäärittely	0	0	0	0	0	0,5	2	0	2	0,5
- Kirjoitus	0	0	0	0	0	0,5	2	0	2	0,5
Toteutus	43	12,5	44	24,25	18	2	44	24,75	149	63,5
- Käyttöliittymän muokkaus	21	9,25	10	3	0	0	0	1	31	13,25
- Entiteettien tyyppeihin liittyvät tehtävät	7	2	0	0	0	0	0	0	7	2
- Käsitemallin tallennus SVG-muodossa	0	0	5	4	1	0	0	0	6	4
- Export all toiminto	0	0	5	3	0	0	0	0	5	3
- Jäsentimen parannus	0	0	0	0	0	0	27	19,25	27	19,25
- GXL tallennuksen muokkaus	0	0	0	1	1	0	5	0	6	1
- Muokauslokin muutokset	3	1,25	0	0	0	0	0	0	3	1,25
- Asennusohje	0	0	5	5,25	3	0	0	0	8	5,25
- Käyttöohje	0	0	5	2	3	0	0	4,5	8	6,5
- Testaus	0	0	2	0	5	2	0	0	7	2
- Ilmenneiden bugien korjaus	5	0	5	6	1	0	5	0	16	6
- Lähdekoodin refaktorointi	5	0	5	0	2	0	5	0	17	0
- Katselmointi	2	0	2	0	2	0	2	0	8	0
Toinen väliesittely	3	3	3	3	6	5	4	3	16	14
- Valmistautuminen	1	1	1	1	3	3	1	0	6	5
- Esitys	2	2	2	2	2	2	2	2	8	8
- Raportin kirjoitus	0	0	0	0	1	0	1	1	2	1
Tunteja yhteensä	55	28,75	55	32	55	32	55	29,5	220	122,25

Kuva A.5: Viidennen iteraation ajankäytön suunnitelma.

A.6 Asiakastapaamiset

Taulukossa A.6 on esitetty projektin asiakastapaamiset. Palavereita oli yhteensä 16, joista 11 olivat virallisia. Viimeisen kahden palaverin kestot ovat arvioita. Palaverien kesto yhteensä on 31h 54min.

Tapaamisen ajankohta	Kesto	Virallinen
Perjantai 15.9.2006 klo 12:10	1h 54min	Kyllä
Perjantai 22.9.2006 klo 12:03	2h 34min	Kyllä
Perjantai 29.9.2006 klo 12:16	2h 45min	Kyllä
Perjantai 6.10.2006 klo 12:04	2h 58min	Kyllä
Perjantai 13.10.2006 klo 12:00	1h 15min	Ei
Perjantai 20.10.2006 klo 12:04	1h 48min	Kyllä
Perjantai 27.10.2006 klo 12:06	1h 35min	Kyllä
Perjantai 3.11.2006 klo 12:05	1h 15min	Ei
Perjantai 10.11.2006 klo 12:05	2h 39min	Kyllä
Perjantai 17.11.2006 klo 12:00	1h	Ei
Perjantai 24.11.2006 klo 12:03	2h 48min	Kyllä
Perjantai 1.12.2006 klo 12:00	1h 40min	Ei
Perjantai 8.12.2006 klo 12:04	1h 36min	Kyllä
Perjantai 15.12.2006 klo 11:25	3h 1min	Kyllä
Tiistai 19.12.2006 klo 15:45	30min	Ei
Keskiviikko 20.12.2006 klo 15:45	2h	Kyllä

Kuva A.6: Projektin asiakastapaamiset

B Termit

Alkuperäinen käyttötapaus	on lähteen sisältämä käyttötapaus.
Core	on sovelluksen ydin, joka ohjaa ohjelman muiden komponenttien toimintaa.
Entiteetti	on vaatimusmäärittelyssä esiintyvä toimija tai toimenpiteen kohde. Käytännössä mikä tahansa substantiivi voi olla entiteetti.
Entiteetin tyyppi	kuvaa entiteetin roolia sovellusalueen käsitelmällin osana.
HeuristicCollection	on luokka, joka säilöö <code>HeuristicModule</code> ja.
HeuristicInterface	on rajapinta, joka määrittää miten <code>HeuristicModule</code> en kanssa kommunikoidaan.
HeuristicModule	on luokka, joka suorittaa heuristiikan sille annetulle jäsenetylle käyttötapaukselle ja palauttaa käsitelmällin.
InputAdapter	on luokka, joka lataa URL:llä osoitetun lähteen sisältämät alkuperäiset käyttötapaukset ja palauttaa ne jäsentämättöminä käyttötapauksina.
InputInterface	on rajapinta, joka määrittää miten <code>InputAdapter</code> in kanssa kommunikoidaan.
InputCollection	on luokka, joka säilöö input adaptereita.
Iteraatio	tarkoittaa yleisesti jonkin asian toistamista uudelleen siten, että edellisen suorituskerran tulos on seuraavan kerran syöte. Sovelluskehityksessä iteraatiolla tarkoitetaan projektin suorittamista pienissä paloissa edellisen iteraation tulosten toimiesä seuraavan iteraation toteutuksen pohjana. Tuloksilla tässä tapauksessa tarkoitetaan kaikkea ohjelmiston kehityksen tuottamaa materiaalia eikä vain

	<p>lähdekoodia. Peräkkäiset iteraatiot eivät välttämättä käytä ollenkaan samaa lähdekoodia vaan koodi voidaan välillä kirjoittaa uudestaan.</p>
Jäsennetty käyttötapaus	<p>on käyttötapaus, jolle on suoritettu morfologinen jäsennys.</p>
Jäsentämätön käyttötapaus	<p>on käyttötapausten suoritusaskeleet tekstimuodossa. Käyttötapaus on jo otettu sisään järjestelmään, mutta sitä ei ole vielä toimitettu parserille.</p>
Käsitemalli	<p>on heuristiikan muodostama malli jäsennetyistä käyttötapausten muodoista.</p>
Käyttötapaus	<p>on kuvaus järjestelmän ja sen käyttäjän välisestä vuorovaikutuksesta tietyn tuloksen aikaansaamiseksi.</p>
Käyttötapausten muoto	<p>kertoo, mitä attribuutteja ja missä järjestyksessä sekä muodossa yksittäisen käyttötapausten kuvaus sisältää. Näitä attribuutteja ovat mm. tiedot pääaktorista ja muista aktoreista, tietoa järjestelmän tilasta ennen ja jälkeen käyttötapausten toiminnan sekä käyttötapausten suoritusaskeleet.</p>
L^AT_EX 2_ε	<p>on ladontaohjelmisto, millä tämäkin dokumentti on tehty.</p>
Moduuli	<p>on ohjelman osa, joka piilottaa varsinaisen toiminnan toteutuksen sisäänsä. Hyvin kirjoitetun moduulin sisäistä toteutusta on helppo muuttaa. Yleensä moduuli toteuttaa jonkin rajapinnan vaatiman toiminnallisuuden. Esimerkiksi UCOT-ohjelmistossa eri lähteistä tullutta dataa voidaan lukea kun vain datan lukemista varten on toteutettu moduuli, joka täyttää UCOT-ohjelmiston syöterajapinnan määrittelyyn.</p>
Output	<p>on luokka, joka hoitaa heuristiikan tuottaman käsitemallin esittämisen/tallentamisen.</p>

OutputCollection	on luokka, joka säilöö Outputeja
OutputInterface	on rajapinta, joka määrittää outputin kanssa kommunikoidaan.
Parser	tarkoittaa morfologista jäsenointiä.
ParserAdapter	on luokka joka toteuttaa ParserInterfacen ja kommunikoi parserin kanssa. Ottaa vastaan jäsenoittämättömän käyttötapauksen ja palauttaa jäsenoitetyn käyttötapauksen.
ParserCollection	on luokka, joka säilöö ParserAdapttereita.
ParserInterface	on rajapinta, joka määrittää miten ParserAdapterin kanssa kommunikoidaan.
Projekti	tarkoittaa tämän dokumentin yhteydessä sovellusprojektiä.
Rajapinta	erottaa kaksi toisistaan erillistä ohjelman osaa toisistaan siten, että osat tietävät vain osan toisen toiminnallisuudesta. Tällöin rajapinnan takan olevaa osaa voidaan vaihtaa toisen osan häiriintymättä.
Sovellusprojekti	on tietotekniikan laitoksen opintojakso.
Spike	on ketterään sovelluskehitykseen liittyvä termi. Se tarkoittaa toteutuskelpoisuuden testaamista. Siinä tehdään yleensä nopea kokeilu jostakin ratkaisusta, jotta sen käyttökelpoisuus selviäisi.
Syöte	on ohjelman vastaanottama data.
Syötemoduuli	on ohjelman osa, joka lukee ohjelmalle tarkoitetun syöteen ja palauttaa ohjelmalle jäsenoitetyn käyttötapauksen. Rakentuu InputAdapterista ja ParserAdapterista.
Tuloste	on ohjelman tuottama data.
Tyyppi	kts. "Entiteetin tyyppi".

UI	tarkoittaa käyttöliittymää (<i>user interface</i>). Tarkentuu myöhemmissä iteraatioissa.
UIInterface	on käyttöliittymän rajapinta.
UCOT	on tämän sovellusprojektin toteuttava ryhmä.
Vaikutussuhde	on kahden entiteetin välillä vallitseva suhde, jossa toinen käyttää toista.