

Kuvatus-sovellusprojekti

Pekka Iso-Ahola

Jussi Perttola

Tommi Tuovinen

Projektisuunnitelma

Julkinen

Versio 1.1.0

28.3.2012

Jyväskylän yliopisto

Tietotekniikan laitos

Jyväskylä

Hyväksyjä	Päivämäärä	Allekirjoitus	Nimenselvennys
Projektipäällikkö	__.__.2012		
Tilaaja	__.__.2012		
Ohjaaja	__.__.2012		

Tietoa dokumentista

Tekijät:

- Pekka Iso-Ahola (PIA) pekka.o.iso-ahola@jyu.fi 040 842 8378
- Jussi Perttola (JP) jussi.t.perttola@jyu.fi 050 412 4420
- Tommi Tuovinen (TT) tommi.s.tuovinen@jyu.fi 040 751 8187

Dokumentin nimi: Kuvatus-projekti, Projektisuunnitelma

Sivumäärä: 32

Tiedosto: kuvatus_projektisuunnitelma_v_1_1_0.tex

Tiivistelmä: Kuvatus-projekti toteuttaa Jyväskylän yliopiston tietotekniikan laitokselle hyperspektrikameran kuvien analysointiin ja oppivien tietokoneohjelmien koulutusaineiston hallintaan soveltuvan työkalun prototyypin. Projektisuunnitelma kuvaa projektin tavoitteet ja suunnitellun läpiviennin. Dokumentissa käsitellään projektin käytänteitä, suunniteltuja vastuualueita ja tehtävien työmääriä, käytettävää prosessimallia, projektin kokonaisuikataulua sekä mahdollisten riskien hallintaa.

Avainsanat: Aikataulu, käytänteet, metatieto, ohjelmistoprojekti, projektin läpiviennin, prosessimalli, prototyyppi, resurssit, riskien hallinta, suunnitelma, tavoitteet, tehtävät, työmäärät, vastualueet.

Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	28.2.2012	Projektisuunnitelman laatiminen aloitettu.	JP
0.0.2	1.3.2012	Johdanto-, termi- ja tavoiteluvut sekä organisaa- tiota ja sen resursseja koskevat luvut alustavasti kirjoitettu.	JP
0.1.0	2.3.2012	Käytänteitä sekä tehtäviä, työmääriä ja vastuu- alueita käsittelevät luvut alustavasti kirjoitettu.	JP
0.1.1	5.3.2012	Koko dokumentti alustavasti laadittu.	JP
0.1.2	5.3.2012	Muotoiluja tarkennettu ja kieliäsvirheitä korjat- tu.	JP
0.1.3	6.3.2012	Ajankäyttösuunnitelmataulukko lisätty, sekä ris- kikuvaukset ja yhteenveto kirjoitettu.	JP
0.1.4	7.3.2012	Muotoiluja tarkennettu ja kieliäsvirheitä korjat- tu.	JP
0.2.0	7.3.2012	Gantt-kaavio lisätty.	JP
0.2.1	8.3.2012	Työtuntitaulukko ladottu L ^A T _E Xilla.	JP
0.2.2	8.3.2012	Lähdekoodiesimerkki lisätty.	JP
0.3.0	15.3.2012	Muotoiluja tarkennettu ja kieliäsvirheitä korjat- tu.	JP
0.3.1	21.3.2012	Kansiorakennetta muutettu, termejä tarkennettu ja kieliäsvirheitä korjattu.	JP
0.4.0	22.3.2012	Tarkennettu kuvaajan määritelmää ja lisätty tar- kennus sovelluksen laskentarajapinnoista.	JP
1.0.0	23.3.2012	Tarkennettu muotoiluja ja viimeistelty kieliäsvir- heitä.	JP
1.1.0	28.3.2012	Viimeistelty kieliäsvirheitä ja tarkennettu muotoiluja.	JP

Tietoa projektista

Kuvatus-projekti toteuttaa Jyväskylän yliopiston tietotekniikan laitokselle hyper-spektrikameran kuvien analysointiin ja oppivien tietokoneohjelmien koulutusaineiston hallintaan soveltuvan WWW-sovelluksen prototyypin.

Tekijät:

- Pekka Iso-Ahola (PIA) pekka.o.iso-ahola@jyu.fi 040 842 8378
- Jussi Perttola (JP) jussi.t.perttola@jyu.fi 050 412 4420
- Tommi Tuovinen (TT) tommi.s.tuovinen@jyu.fi 040 751 8187

Tilaja:

- Paavo Nieminen paavo.j.nieminen@jyu.fi 040 576 8507
- Marko Peltola marko.peltola@jyu.fi 041 449 8622
- Ilkka Pölönen ilkka.polonen@jyu.fi 0400 248 140
- Tero Tuovinen tttuovin@mit.jyu.fi 040 531 0791

Ohjaajat:

- Tero Hänninen tero.j.hanninen@jyu.fi 0400 240 468
- Jukka-Pekka Santanen santanen@mit.jyu.fi 050 550 4666

Yhteystiedot:

- Sähköpostilista kuvatus@korppi.jyu.fi
- Sähköpostiarkisto <https://korppi.jyu.fi/kotka/servlet/list-archive/kuvatus/>
- Sähköpostilista (opetus) kuvatus_opetus@korppi.jyu.fi
- Sähköpostiarkisto (opetus) https://korppi.jyu.fi/kotka/servlet/list-archive/kuvatus_opetus/
- Työhuone AgC 223.2

Sisältö

1	Johdanto	1
2	Termejä	3
2.1	Aihealueen termejä	3
2.2	Kehitysvälineisiin ja -tekniikoihin liittyviä termejä	4
3	Tavoitteet	5
3.1	Taustaa ja tarpeita	5
3.2	Toteutettava sovellus	5
3.3	Tulokset	7
3.4	Oppimistavoitteet	8
4	Organisaatio ja resurssit	11
4.1	Projektioorganisaatio	11
4.2	Projektin tilat ja laitteet	11
4.3	Ohjelmointi- ja dokumentointityökalut	12
4.4	Luennot ja perehdytykset	13
5	Käytänteet	14
5.1	Kokoukset	14
5.2	Tiedotus	15
5.3	Tiedostojen nimeäminen ja hakemistorakenne	16
5.4	Lähdekoodi	18
5.5	Testaus	18
5.6	Versiohallinta ja -numerointi	19
5.7	Katselmoinnit ja tulosten hyväksyminen	20
5.8	Tulosten koostaminen ja toimittaminen	20
6	Tehtävät, työmäärät ja vastualueet	21
6.1	Projektipäällikkö ja varapäällikkö	21
6.2	Vastualueet tulosten osalta	21
6.3	Tehtäväkohtaiset työmäärät ja työnjako	22
7	Prosessimalli ja aikataulu	24
7.1	Prosessimalli	24
7.2	Aikataulu	24

8 Riskien hallinta	26
8.1 Arvioidut riskit, niiden todennäköisyydet ja haittavaikutukset	26
8.2 Sidosryhmien toiminnan viiveet	26
8.3 Puutteet projektiryhmän tietotaidoissa	27
8.4 Muutostarpeet projektiryhmän tietokoneissa	28
8.5 Jäsenten odottamattomat poissaolot	28
8.6 Projektin hallinnan puutteet	29
8.7 Valitun alustan asettamat rajoitteet	29
9 Yhteenveto	31
10 Lähteet	32

1 Johdanto

Viime vuosina Jyväskylän yliopiston tietotekniikan laitoksen tutkimusryhmissä on alettu analysoida ja käsitellä hyperspektrikuvia, sekä muodostaa saatujen tulosten perusteella oppivien tietokoneohjelmien koulutusaineistoja, joita voidaan hyödyntää jatkotutkimuksissa. Hyperspektrikuvat ovat hyperspektrikameralla otettuja, eri aallonpituuksia esittävistä kuvista koostuvia kuvasarjoja. Kuvatus-projektissa kehitettävää sovellusta tullaan käyttämään työkaluna hyperspektrikuvien analysoinnissa.

Kuvatus-projekti toteuttaa toiminnallisen prototyypin kuva-analyysimenetelmissä käytettävän opetusdatan **valitsemiseen ja hallintaan** käytettävästä sovelluksesta. Kehitettävällä sovelluksella tulee pystyä määrittämään oppivien tietokoneohjelmien koulutusaineistoja rajaamalla alueita hyperspektrikuvista ja määrittelemään tehdyille rajauksille luokkia. Tämän lisäksi sovelluksella tulee voida hallita näitä koulutuseli opetusaineistoja.

Tietotekniikan laitoksen tutkimusryhmien käyttämät hyperspektrikuvat säilytetään yhteisessä YouData-tietojärjestelmässä. Kuvatus-projektissa kehitettävän sovelluksen tulee toimia saumattomasti YouDatan sekä siellä säilytettävien aineistojen ja niihin liittyvien projektiryhmien kanssa.

Sovelluksen taustalla olevana kokonaistavoitteena on tehostaa yliopistolla tehtävää tutkimusta ja yhtenäistää laitosten välisiä käytäntöjä hyperspektrikuvien analysoinnissa. Käytännössä tämä edellyttää, että kehitettävällä sovelluksella tulee olla helpompi ja nopeampi tehdä hyperspektrikuviin valintoja ja muodostaa valinnoista opetusaineistoja kuin aiemmin käytössä olleilla työkaluilla.

Projektiryhmä laatii projektisuunnitelman lisäksi muita projektia ja sovellusta määritteleviä dokumentteja. Sovelluksen teknistä toteutusta kuvaavassa sovellusraportissa kuvataan yleisellä tasolla sovelluksen käyttöliittymän ja ohjelmakoodin toteutusratkaisut. Lisäksi sovellusraportissa esitellään mahdolliset havaitut ongelmat ja jatkokehitysehdotukset. Projektiraportissa kuvataan projektin läpiviennin toteuma ja analysoidaan sovelluksen läpiviennin onnistumista. Projektiraportissa myös verrataan projektin toteumaa suunnitelmaan sekä pohditaan mahdollisten erojen syitä ja vaikutuksia.

Projektisuunnitelma kuvaa yksityiskohtaisesti projektin suunniteltua läpivientä ja määrittelee sille asetetut tavoitteet. Projektisuunnitelmassa esitellään lisäksi kaik-

ki projektiorganisaatioon kuuluvat henkilöt ja muut projektiin keskeisesti liittyvät resurssit. Lisäksi käsitellään projektissa sovellettavia käytänteitä, ryhmän jäsenille osoitettuja tehtäviä ja vastuualueita sekä mahdollisten riskien hallintaa. Lisäksi dokumentissa esitetään suunnitelma ryhmän jäsenten henkilökohtaisista työmääristä ja projektin kokonaisaikataulusta. Projektisuunnitelmaa laadittaessa on käytetty mallina Judo-projektin projektisuunnitelmaa [7].

Luvussa 2 kuvataan projektiin ja toteutettavaan sovellukseen liittyviä termejä. Luku 3 keskittyy esittelemään projektin tavoitteita sovelluksen ohella myös muiden tulosten ja jäsenten oppimisen osalta. Luvussa 4 käsitellään projektiorganisaatiota, sovellusprojektissa käytössä olevia resursseja ja ryhmän jäsenille tarjottavaa opetusta. Luku 5 käsittelee projektin käytänteitä muun muassa kokousten, tiedotuksen, tiedostojen nimeämisen ja dokumentoinnin osalta. Luku 6 käsittelee projektiryhmän tehtäviä ja niiden työmääriä sekä ryhmän jäsenten henkilökohtaisia vastuualueita. Luvussa 7 kuvataan projektissa käytettävää prosessimallia sekä projektin kokonaisaikataulua. Luku 8 keskittyy mahdollisiin projektin läpivientiin vaikuttaviin riskeihin, niiden vaikutuksiin ja hallintaan.

2 Termejä

Luvussa esitellään dokumentissa käytettäviä aihealueen ja kehitysvälineiden termejä.

2.1 Aihealueen termejä

Dokumentissa esiintyvät projektin hallintaan ja aihealueeseen liittyvät termit ovat seuraavat:

Hyperspektrikuva	on hyperspektrikameralla otettu kuva, joka koostuu useista, sähkömagneettisen säteilyn eri aallonpituusalueita kuvaavista kuvista.
Ketterä ohjelmistokehitys	sisältää useita ohjelmistotuotantoprojektin läpiviennessä käytettäviä menetelmiä, jotka tunnustavat <i>Agile Manifestossa</i> [1] esiteltyjä arvoja. Ketterän ohjelmistokehityksen peruseriaatteisiin kuuluu toimivan ohjelmiston priorisointi, nopea muutokseen reagointi sekä joustava viestintä kehittäjän ja asiakkaan välillä.
Metatieto	on tiedoston sisältöä kuvailevaa tietoa, joka helpottaa tiedon etsintää ja lajittelua. Tiedoston metatietoihin voi kuulua esimerkiksi käyttö- ja muutoshistoriatiedot sekä kuvaus sen suhteesta muihin tiedostoihin.
Opetusaineisto	eli oppivien tietokoneohjelmien koulutusaineisto on joukko kerättyä aineistoa, jonka avulla voidaan kouluttaa luokitin tunnistamaan samoja piirteitä mistä tahansa uudesta aineistosta.
Prototyyppi	on osittaista toiminnallisuutta sisältävä versio tuotantokäyttöön tarkoitettu tuotteesta, jolla pyritään keräämään käyttäjäkokemuksia.

WWW-käyttöliittymä on WWW-selaimella käytettävä sovelluksen käyttöliittymä.

2.2 Kehitysvälineisiin ja -tekniikoihin liittyviä termejä

Dokumentissa esiintyy seuraavia kehitysvälineisiin ja -tekniikoihin liittyviä termejä:

JavaScript on oliopohjainen komentosarja- eli skriptikieli, jonka avulla voidaan luoda dynaamista sisältöä WWW-sivuille.

JIRA on ohjelmisto, jota projektissa käytetään sovelluksen vaatimusten ja virheiden määrittelyyn ja hallintaan.

jQuery on avoimen lähdekoodin lisenssin alainen JavaScript-kirjasto, jota käytetään WWW-käyttöliittymän dynaamisten ominaisuuksien määrittelyyn.

Korppi on Jyväskylän yliopistossa kehitetty opintotietojärjestelmä.

MongoDB on avoimen lähdekoodin lisenssin alainen NoSQL-dokumentti-tietokanta.

NoSQL tarkoittaa perinteisestä relaatiotietokantamallista poikkeavia tietokantoja.

Ruby on avoimen lähdekoodin lisenssin alainen olio-ohjelmointikieli. Rubyssa on vahva ja dynaaminen tyyppitys, ja se on suoritusmalliltaan tulkittava [5].

Ruby on Rails on avoimen lähdekoodin WWW-sovelluskehys Ruby-ohjelmointikielille. Ruby on Rails käyttää MVC-arkkitehtuurimallia, jossa ohjelma jaetaan malleihin, näkymiin ja ohjaimiin.

YouData on Judo-projektissa kehitetty tiedostojen jakamiseen käytettävä tietojärjestelmä.

YouSource on Verso-projektissa kehitetty lähdekoodien julkaisujärjestelmä, jota käytetään projektin tulosten julkistamiseen ja versiohallintaan.

3 Tavoitteet

Luvussa käsitellään projektissa toteutettavalle sovellukselle ja muille tuloksille sekä ryhmän oppimiselle asetettuja tavoitteita.

3.1 Taustaa ja tarpeita

Hyperspektrikuvat ovat hyperspektrikameralla otettuja, eri aallonpituuksia esittäviä kuvista koostuvia kuvasarjoja. Hyperspektrikuvia voidaan käyttää apuna esimerkiksi metsän biomassan arvioinnissa, paperiteollisuuden laadunvalvonnassa, rikospaikkatutkimuksessa ja ihotautien havaitsemisessa. Analysoimalla useita samaan aihealueeseen liittyvien hyperspektrikuvien piirteitä voidaan koota oppivien tietokoneohjelmien koulutusaineistoja, joiden avulla voidaan esimerkiksi opettaa tietokone tunnistamaan automaattisesti sairas ihoalue terveestä.

Jyväskylän yliopiston tietotekniikan laitos on osallisena useassa tutkimusprojektissa, joissa tutkitaan hyperspektrikameralla kerättävän aineiston erilaisia käytännön sovelluksia. Tällä hetkellä laitoksella ei ole käytössä käyttäjien tarpeita vastaavaa ohjelmistoa kuva-analyysimenetelmissä käytettävän opetusaineiston valitsemiseen ja hallintaan, joten hyperspektrikuvien käsittely on hyvin työlästä. Osittain tästä syystä myöskään tutkijoiden työskentelykäytänteet opetusdatan valitsemisessa ja hallinnassa eivät ole välttämättä yhtenevät edes tutkimusryhmän sisällä. Projektin tilaajana toimiva, Jyväskylän yliopiston tietotekniikan laitos toivoo, että Kuvatus-projektissa kehitettävä sovellus auttaisi yhtenäistämään käytänteitä myös eri tutkimusryhmien välillä. Lisäksi sovelluksen avulla laitoksen tutkimusta voidaan havainnollistaa yhteistyökumppaneille ja opiskelijoille.

3.2 Toteutettava sovellus

Kuvatus-projekti toteuttaa Jyväskylän yliopiston tietotekniikan laitokselle toiminnallisen prototyypin kuva-analyysimenetelmissä käytettävän opetusaineiston valitsemiseen ja hallintaan tarkoitetusta WWW-sovelluksesta. Kehitettävän sovelluksen kautta tulee päästä käsiksi YouData-tietojärjestelmässä säilytettäviin **hyperspektrikuviin**, niihin **liittyviin valintoihin** ja **valintaluokkiin**, sekä niiden avulla muodostettaviin **opetusaineistoihin** ja **projekteihin**.

Sovelluksessa tulee pystyä rajaamaan alueita hyperspektrikuvista ja määrittelemään tehdyille valinnoille valintaluokkia. Esimerkiksi ihonäytteitä analysoiva käyttäjä saattaa haluta merkitä kuvaan kolme siinä näkyvää ihotyyppeä ja osoittaa niiden esiintymät merkkamalla vastaavat alueet jollain **rajaustyökalulla**. Jokainen kuvista esittävistä ihotyyppeistä on siten yksi valintaluokka, ja yksittäiseen kuvaan merkityt tietyt ihotyypin esiintymät muodostavat yhdessä valinnan.

Yhdestä tai useammasta valintaluokasta tulee voida muodostaa opetusaineisto, jota voi hyödyntää sovelluksen ulkopuolella. Sovellukseen tulee myös toteuttaa rajapinta, jonka kautta opetusaineiston pystyy välittämään kolmannen osapuolen **laskentaohjelmaan** analysoitavaksi.

Sovelluksen tärkein ominaisuus on monipuolinen **rajaustyökalu**, jolla käyttäjä voi valita alueita ja ryhmitellä tehtyjä valintoja tähän tarkoitukseen suunnitellussa käyttöliittymänäkymässä. Nykyisin käytössä olevissa sovelluksissa erityisesti valinnan tekeminen on hankalaa ja hidastaa koko työskentelyprosessia. Tilaaja toivoo, että sovelluksessa on ainakin suorakulmio- ja ympyrä- tai ellipsityökalu. Lisäksi tutkitaan mahdollisuutta toteuttaa älykäs valintatyökalu, joka osaa valita samanvärisiä alueita valitun pisteen ympäriltä. Sovellus piirtää käsiteltävästä hyperspektrikuvasta kuvaajan, joka kuvaa yksittäisen pikselin intensiteetin eri aallonpituuksilla.

Ohjelmassa tulee olla myös mahdollisuus **lähentää ja loitontaa** käsiteltävää hyperspektrikuvaa kuitenkin niin, ettei kuvan skaalauksessa tapahdu tiedon katoamista. Tämä tarkoittaa käytännössä sitä, että koon muunnos tulee tehdä kokonaislukumonikertoina. Käyttäjän pitää myös pystyä helposti valitsemaan mitä hyperspektrikuvan aallonpituutta tarkastellaan.

Tilaaja määritteli olennaiseksi tavoitteeksi sujuvan **yhteistoiminnan YouData-tiedonjakopalvelun kanssa**, johon Kuvatus-projektissa toteutettava sovellus ainakin osittain integroidaan. Lähinnä YouDatan toteutuksesta johtuen tilaaja haluaa projektissa kehitettävään ohjelmaan WWW-käyttöliittymän. Tällöin sovelluksen käyttö ei myöskään edellytä käyttäjältä ohjelmiston asennusta.

Sovellus **integroidaan YouDataan** siten, että samoja projektiryhmiä ja niiden tiedostoja voidaan hyödyntää molemmissa sovelluksissa. Tunnistautuminen järjestelmään tapahtuu samalla tavalla Korppi-tunnuksilla kuin YouDatassa. Integraation ensisijaisena tarkoituksena on parantaa ohjelmien yhteiskäyttöä ja vähentää päällekkäisiä käyttäjäryhmiä, mutta se todennäköisesti parantaa samalla myös kehitettävän sovelluksen suorituskykyä erityisesti tiedostojen lataamisen ja tallentamisen osalta.

Yhteen projektiin voi liittyä useita hyperspektrikuvia, joihin voi liittyä useita valintaluokkia ja valintoja. Kuviin ja tehtyihin valintoihin liittyvät **metatiedot** määrittelevät, miten kuvat, valintaluokat ja valinnat liittyvät toisiinsa. Lisäksi jokaiseen kuvaan tallennetaan metatietona muutoshistoria, joka sisältää tiedon siitä, millaiset valinnat liittyvät kuhunkin versioon. Tällä tavoin käyttäjä voi palauttaa aiemmin tekemänsä valinnat helposti takaisin.

Sovelluksen **ohjelmointikieleksi** valittiin **Ruby on Rails** pääosin siksi, että sitä on käytetty myös YouDatan toteutuksessa. Ettei sovellus olisi riippuvainen käyttöjärjestelmästä, sovittiin, että projektiryhmän tulee pystyä osoittamaan ainakin yksi **Mozilla Firefoxin** versio, jolla ohjelma toimii käyttöjärjestelmästä riippumatta. Tilaajan toiveen mukaisesti selaimesta valitaan sellainen versio, jonka voi ajaa suoraan esimerkiksi massamuistilta. Tämä helpottaa huomattavasti sovelluksen käyttöä tavallisen työympäristön ulkopuolella, esimerkiksi asiakkaan tietokoneella.

Kuvatus-projektin päätavoite on toimittaa tilaajalle toimiva sovellus, jossa on huomioitu esitettyjen vaatimusten ohella sovelluksen laajennettavuus. Käytännössä tämä tarkoittaa sitä, että vaikkei kaikkia ominaisuuksia ehdittäisi toteuttaa, varataan niille kuitenkin tilaa käyttöliittymässä ja toteutetaan lähdekoodiin alustavat aliohjelmarungot.

Koska valmis sovellus toimitetaan tilaajalle jo huhti-toukokuun vaihteessa, projektiryhmän jäsenet ja tilaaja ovat yhtä mieltä siitä, ettei kaikkia tilaajan toivomia ominaisuuksia todennäköisesti ehditä toteuttaa kokonaan. Projektiryhmä pitää tilaajan ajan tasalla projektin etenemisestä, jolloin tilaajan on helpompi seurata prosessin etenemistä. Näin tilaajan edustajat pystyvät paremmin osoittamaan ne sovellukselle asetetut vaatimukset, joiden toteuttaminen on ensisijaista.

3.3 Tulokset

Sovelluksen prototyypin ohella projektiryhmä toteuttaa seuraavat tulokset:

- **Ajankäyttöraportti** sisältää ryhmän jäsenten kirjaamat työtunnit sekä niiden jakautumisen eri tehtäville ja tehtäväkokonaisuuksille.
- **Asennusohje** neuvoo järjestelmän asennuksen ja käyttöönoton palvelimelle.
- **Esittelymateriaalit** sisältävät väli- ja loppuesittelyn esitysgrafiikat ja muistiot.
- **Itsearviointit** sisältävät ryhmän jäsenten arviointit omasta, ryhmän, tilaajan edustajien, ohjaajien ja atk-tuen toiminnasta ja onnistumisesta sekä luennoista

- ja perehdytyksistä.
- **Kokousten dokumentit** sisältävät kokousten esityslistat, pöytäkirjat ja niissä esitetyt tilakatsaukset.
 - **Käyttöohje** neuvoo järjestelmän käytön WWW-käyttöliittymän sekä erityisesti valintatyökalun osalta.
 - **Lähdekoodi** sisältää projektissa toteutetun sovelluksen kommentoidun lähdekoodin.
 - **Oheiskurssin materiaalit** sisältävät oheiskurssien suoritukseen kuuluvat kirjoitusharjoitukset ja muun materiaalin.
 - **Projektiraportti** dokumentoi projektin toteutuneen läpiviennin ja vertaa toteumaa suunnitelmaan sekä arvioi erojen syitä ja vaikutuksia.
 - **Projektisuunnitelma** on projektin läpivientisuunnitelma.
 - **Sovellusraportti** kuvaa projektissa toteutetun sovelluksen rakenteen ja toiminnot sekä mahdolliset puutteet ja jatkokehitysideoita.
 - **Sähköpostiarkisto** sisältää kaikki projektiorganisaation sähköpostilistalle lähetetyt viestit.
 - **Testaussuunnitelma** kertoo, miten sovelluksen järjestelmätestaus toteutetaan.
 - **Testausraportit** esittävät, miten sovelluksen järjestelmätestaukset suoritettiin, mitä virheitä testauksilla löydettiin ja miten ne tulivat esille.
 - **Vaativuusmäärittely** kuvaa sovelluksen tavoitteet, tekniset ja toiminnalliset vaatimukset sekä rajoitteet.

Kaikki projektidokumentit sijoitetaan Creative Commons Attribution-ShareAlike 3.0 Unported (cc-by-sa 3.0) -lisenssin alaisuuteen. Kaikki projektin aikana toteutettu lähdekoodi sijoitetaan avoimen lähdekoodin MIT-lisenssin alaisuuteen.

3.4 Oppimistavoitteet

Sovellusprojekti-opintojakson päätavoitteena on totuttaa opiskelijat projektimuotoiseen työskentelyyn. Jokainen projektiryhmän jäsen saa kokemusta useilta ohjelmistokehityksen eri osa-alueilta, sillä pieni ryhmäkoko pakottaa jakamaan useita tehtäviä samoille henkilöille. Erityisesti projektipäällikkö oppii projektityöskentelyn suunnittelussa ja hallinnassa vaadittuja johtamistaitoja ja ajankäytön hallintaa. Kaikki ryhmän jäsenet oppivat monipuolisesti taitoja ohjelmiston määrittelyyn, suunnitteluun, toteutukseen ja testaukseen liittyen, minkä lisäksi tiiviissä ryhmässä kaikki oppivat väistämättä ryhmätyötaitoja.

Viikoittaiset kokoukset ovat olennainen osa sovellusprojektia ja sitä kautta myös oppimisprosessia. Kokouksissa ryhmän jäsenet oppivat noudattamaan asianmukaisia kokouskäytänteitä, minkä lisäksi he oppivat laatimaan esityslistoja ja pöytäkirjoja. Kaikki ryhmän jäsenet toimivat vuorollaan myös puheenjohtajan ja sihteerin roolissa, joten jokainen saa kokemusta myös kyseisistä tehtävistä.

Sovellusprojekti kurssin yhteyteen on liitetty opiskelijoiden puhe- ja kirjoitusviestintätaitoja kehittävä *Projektiviestintä IT-alalla* -kurssi, johon kuuluu erilaisia kirjoitus- ja esiintymisharjoituksia. Myös kaikista sovellusprojektin aikana laadittavista dokumenteista saatava palaute tukee viestintätaitojen oppimista. Projektiin liittyy myös oheiskurssi *Sovellusprojektin hallintaa, viestintää ja työkaluja*. Tämän kurssin aikana opitaan projektityöskentelyn edellyttämiä taitoja, ryhmätyöskentelyä ja suunnitelmallisuutta. Kurssin aikana opitaan myös, mitä käytettävyyks käytännössä tarkoittaa.

Projektin aikana opiskelijat kohtaavat monia odottamattomia tilanteita. Aiempien sovellusprojektiryhmien materiaalien läpikäynti opettaa hyväksi havaittuja käytänteitä, mutta näyttää myös sen, miten joitain asioita ei välttämättä kannata tehdä. Vastaavasti projektiryhmä oppii sosiaalisia taitoja neuvotellessaan asiakkaan kanssa sovelluksen toiminnallisuuksista ja pyrkiessään kaikkia osapuolia miellyttäviin ratkaisuihin. Myöskään ryhmän sisäisessä työskentelyssä vaadittujen viestintätaitojen vaikutusta oppimiseen ei voida jättää huomiotta.

Kaikki sovellusprojektiryhmän jäsenet oppivat myös **ohjelmointitaitoja** erityisesti, jos käytetyt ohjelmointikielet tai sovelluskehikset eivät ole entuudestaan tuttuja. Ohjelmoinnin yhteydessä ryhmän jäsenet oppivat väistämättä myös lukemaan toisten kirjoittamaa lähdekoodia ja hyödyntämään versiohallintaohjelmistoa, joka mahdollistaa koodin yhtäaikaisen kehityksen. Sovelluksen vaatimusmäärittelyn yhteydessä käytetään JIRA-sovellusta, jonka avulla jokainen ryhmän jäsen oppii kirjamaan vaatimuksia ja niiden tiloja, virheitä sekä kehitysideoita.

Projektityöskentely edellyttää kaikilta ryhmän jäseniltä tiivistä **yhteistyötä**, sillä projektin läpivienti riippuu kaikkien jäsenten panoksesta. Ryhmässä työskenteleminen vaatii jäsenten väliltä luottamusta ja kykyä ottaa vastuuta. Projektityöskentelyssä korostuvat myös aloitekyky ja omatoimisuus, mutta toisaalta jokaisen tulee samalla pystyä pitämään muut ryhmän jäsenet ajan tasalla siitä, mitä kulloinkin tekee.

Edellä mainittujen oppimistavoitteiden lisäksi projektiryhmän jäsenet ovat asettaneet itselleen **henkilökohtaiset oppimistavoitteet**:

- Pekka Iso-Ahola asettaa oppimistavoitteikseen projektitoiminnan käytänteiden ja ryhmätyötaitojen oppimisen sekä on kiinnostunut oppimaan WWW-sovelluksen kehittämiseen liittyviä taitoja.
- Jussi Perttolan tavoitteena on oppia projektiryhmän johtamisessa vaadittavia taitoja ja organisointikykyä. Lisäksi hän haluaa osallistua sovelluksen käyttöliittymän suunnitteluun, kehittää ohjelmointivarmuuttaan sekä oppia Ruby on Rails -sovelluskehityksen käyttöä ja Ruby-ohjelmointia.
- Tommi Tuovinen haluaa oppia työelämässä vaadittuja projektityöskentely- ja ryhmätyötaitoja. Hän haluaa myös käytännön kokemusta ohjelmistokehityksestä sekä sen kaikista osa-alueista ja niihin liittyvistä tehtävistä.

4 Organisaatio ja resurssit

Luvussa käsitellään projektiorganisaatioon kuuluvien henkilöiden lisäksi muita projektin käytössä olevia resursseja, kuten työtiloja, laitteita ja työkaluja.

4.1 Projektiorganisaatio

Projektiryhmään kuuluu kolme **jäsentä**: Pekka Iso-Ahola, Jussi Perttola ja Tommi Tuovinen. Jussi Perttola toimii projektipäällikkönä ja Pekka Iso-Ahola varapäällikkönä. Projektityöskentelyssä hyödynnetään kunkin ryhmän jäsenen henkilökohtaisia taitoja ja osaamista. Pekka Iso-Ahola on teknisesti kyvykäs ohjelmoija, joka osaa tarttua nopeasti uusiin tehtäviin. Jussi Perttolalla on aiempaa kokemusta projektipäällikkönä toimimisesta. Hän osaa tarvittaessa puuttua ongelmakohtiin sekä antaa rakentavaa kritiikkiä. Tommi Tuovisen erityisosaamiseen kuuluvat laajojen kokonaisuuksien hahmotuskyky sekä matemaattisen analysoiva lähestymistapa.

Tilaaajan edustajina toimivat Paavo Nieminen, Marko Peltola, Ilkka Pölönen ja Tero Tuovinen, jotka työskentelevät tietotekniikan laitoksella. Projektin vastaavana **ohjaajana** toimii tietotekniikan laitokselta Jukka-Pekka Santanen ja teknisenä ohjaajana Tero Hänninen.

Projektin sidosryhmien osalta Jyväskylän yliopiston tietohallintokeskus vastaa ryhmän käytössä olevien laitteistojen ja ohjelmistojen ylläpidosta. Projektiryhmän yhteyshenkilönä ATK-lähituessa toimii Santeri Lapinmäki.

Oheiskursseilla puheviestintään liittyvissä kysymyksissä neuvoo Minna Haapsaari ja **kirjalliseen viestintään** liittyviin kysymyksiin vastaa Kaisa Leino. Käytettävyyden osalta ryhmää neuvoo Meeri Mäntylä.

4.2 Projektin tilat ja laitteet

Projektiryhmän työtilana toimii **projektihuone** Ag C223.2, joka sijaitsee Agoran C-siiven toisessa kerroksessa sovellusprojektien tiloissa. Huoneen solussa on monitoimilaite, jota projektiryhmän jäsenet voivat käyttää tulostukseen, kopiointiin ja skannaamiseen.

Projektin aikana ryhmä käyttää **kokoustilana** kokoushuonetta Ag C226.2. Kokoushuoneessa olevan tietokoneen ja videoprojektorin lisäksi projektiryhmä saa tarvittaessa kokouskäyttöön kannettavan tietokoneen.

Projektiryhmällä on käytössä kolme **tietokonetta**, joista kahdessa on käyttöjärjestelmänä Linux Fedora 14 ja yhdessä Windows 7 Enterprise. Jäsenten käytössä on myös yksi tietohallintokeskuksen virtuaalipalvelin, jossa sovelluksen toimintaa voidaan testata.

Projektiryhmän käytössä on kaksi **verkkolevyä**. Toinen verkkolevy on tarkoitettu ryhmän sisäiseen tiedostojen jakamiseen ja toiselle tulevat projektin WWW-sivut.

4.3 Ohjelmointi- ja dokumentointityökalut

Ohjelmointikielenä sovellusprojektissa käytetään Rubya ja sovelluskehityksenä Ruby on Railsia. WWW-käyttöliittymän toteutuksessa hyödynnetään Java-Scriptiä ja **HTML5:ttä** sekä erityisesti sen kuvien käsittelyyn soveltuva *Canvas*-elementtiä. Tietojen tallentamiseen käytetään MongoDB-nimistä NoSQL-tietokantaa, joka on käytössä myös YouDatassa. Sovellukseen kirjautumiseen käytetään Korppi-tunnuksia, joita varten toteutetaan **Korppi-tietojärjestelmän** kirjautumisrajapinnan toteutettava komponentti. Ruby on Rails -sovelluksen kehittämisessä ei käytetä mitään tiettyä sovelluskehitysympäristöä, vaan jokainen ryhmän jäsen voi käyttää parhaaksi katsomaansa editoria tai sovelluskehitysympäristöä.

Tulosten versiohallintaan käytetään Git-versiohallintaohjelmistoa ja YouSource-nimistä lähdekoodien julkaisujärjestelmää. Luokkadokumentaatio muodostetaan **lähdekoodeista** RDoc-työkalulla.

Projektin keskeisimmät ja laajimmat **dokumentit** kirjoitetaan \LaTeX -ohjelmistolla. Muiden dokumenttien laatimiseen ryhmän jäsenet voivat käyttää kulloiseenkin tilanteeseen parhaiten soveltuva tekstinkäsittelyohjelmistoa.

Toteutettavan **sovelluksen vaatimusten** suunnittelussa ja hallinnassa hyödynnetään JIRA-tikettijärjestelmää. JIRAan kirjataan sovelluksen vaatimuksia ja virheitä, kiinnitetään ne jollekin jäsenelle sekä seurataan työn edistymistä. Myös asiakkaan edustajat voivat seurata tehtäväkokonaisuuksien edistymistä JIRA-järjestelmän avulla ja lisätä tarvittaessa uusia vaatimuksia.

Sovellusprojektiin käytettävien työtuntien kirjaamisen käytetään Petri Heinosen ke-

hittämää Excel-pohjaista ajankäytönseurantasovellusta [3]. Sen avulla muodostettu- ja kuvaajia käytetään myös projektin tilakatsauksissa.

4.4 Luennot ja perehdytykset

Sovellusprojektiurssin ohella opiskelijat suorittavat oheiskurssit *Sovellusprojektin hallintaa, viestintää ja työkaluja* sekä *Projektiviestintä IT-alalla*. Näillä kursseilla opiskelijat oppivat projektin hallintaan ja projektiviestintään liittyviä taitoja. Luennot ja tapaamiset keskittyvät seuraaviin aiheisiin:

- kokous- ja neuvottelukäytänteet (Haapsaari)
- esittely ja esiintyminen (Haapsaari)
- kirjoitusviestintä (Leino)
- projektin johtaminen ja hallinta (Santanen)
- mahdolliset projektipäälliköiden tapaamiset (Santanen)
- käytettävyyden luennot ja ryhmätyöt (Mäntylä)
- tekijänoikeus ja sopimukset (Santanen)
- WWW-ohjelmointi HTML5:n ja JavaScriptin avulla (Vepsäläinen)
- Git-versiohallinnan ja YouSourcen käyttö (Valkama)
- Ruby on Railsin perusteet (Hänninen)
- kaksi väliesittelyä (Haapsaari ja Santanen).

Projektiryhmän jäsenet käyttävät projektin aikana erilaisia työkaluja ja tekniikoita, jotka eivät ole heille entuudestaan tuttuja. Näiden työkalujen käyttöä, erityisesti Ruby-ohjelmointikieltä ja Ruby on Rails -sovelluskehystä ryhmän jäsenet opiskelevat omatoimisesti Internetistä löytyvistä oppaista. Lisäksi opiskelijoiden käytössä on Agile Web Development with Rails -kirja [4].

5 Käytänteet

Luvussa kuvataan projektissa noudatettavia käytänteitä. Nämä käytänteet edesauttavat projektin läpivientiä, asetettujen tavoitteiden saavuttamista ja tulosten toteuttamista. Käytänteiden tarkoitus on varmistaa, että projekti etenee aikataulussa ja sen aikana syntyvät tulokset ovat korkealaatuisia.

5.1 Kokoukset

Projektiorganisaatio pitää viikoittaisen kokouksen torstaisin. Tähän tekevät poikkeuksen viikot 8 ja 10, jolloin kokousta ei järjestetä projektiorganisaation jäsenten muista sitoumuksista johtuen. **Kokouksissa käsitellään** kuluneen viikon aikana tapahtuneita projektin etenemiseen vaikuttaneita asioita sekä tulevia toimenpiteitä ja tarvittavia päätöksiä. Jokaisessa kokouksessa käydään myös läpi edellisen kokouksen pöytäkirjaan merkityt päätökset sekä osallistujille osoitetut tehtävät ja niiden tila.

Projektipäällikkö esittelee ensimmäisten viikkojen jälkeen kokouksissa viikoittain **tilakatsauksen**, jossa kuvataan mitä projektissa on tehty viimeisen viikon aikana, mitä ongelmia on kohdattu ja mitä tehdään seuraavaksi. Tilakatsauksessa esitellään, miten projektiryhmän ja sen yksittäisten jäsenten käyttämät työtunnit jakautuvat projektin eri tehtäväkokonaisuuksien välillä sekä kuinka monta tuntia ryhmä on käyttänyt projektiin eri viikkoina.

Kokouksissa keskustellaan toteutettavan sovelluksen ominaisuuksista ja vaatimuksista sekä niiden toteutusratkaisuista. Kokouksissa käsiteltävät asiat pyritään käymään läpi niin perusteellisesti, että asiakkaan edustajat ja projektiryhmän jäsenet ymmärtävät asiat samalla tavalla, eikä väärinymmärryksiä pääse syntymään. Jos projektiryhmällä on esittää sovelluksesta konkreettisia käyttöliittymädemonstraatioita tai prototyypppejä, ne esitetään viikoittaisessa kokouksessa. Kokouksissa sovietaan myös projektin läpivientiin liittyvistä käytänteistä, kuten sovelluksen ja dokumentaation sijoittamisesta avoimen lähdekoodin lisenssin alle.

Jokainen ryhmän jäsen toimii vuorollaan **sihteerinä** ja **puheenjohtajana**. Opiskelijat toimivat samassa roolissa kaksi peräkkäistä kokouskertaa, jonka jälkeen sihteerinä toiminut projektiryhmän jäsen ottaa puheenjohtajan roolin. Puheenjohtaja johtaa keskustelua ja pitää huolen siitä, että kokous etenee esityslistan osoittamalla taval-

la. Sihteerin tehtävä on laatia kokouksesta **pöytäkirja**, jonka puheenjohtaja tarkistaa ennen sen toimittamista projektiorganisaation jäsenille. Jokainen ryhmän jäsen toimittaa myös ensimmäisen laatimansa pöytäkirjan kirjoitusviestinnän opettaja Kaisa Leinon ja vastaavana ohjaajana toimivan Jukka-Pekka Santasen tarkastettavaksi. Pöytäkirjaan voidaan esittää muutoksia joko etukäteen tai seuraavassa kokouksessa, jossa pöytäkirjaa käsitellään. Kokouksessa pöytäkirja voidaan joko hyväksyä sellaisenaan, hyväksyä muutoksin tai jättää hyväksymättä.

5.2 Tiedotus

Projektiin liittyvien asioiden tiedotuksesta projektiryhmän ja muun projektiorganisaation välillä vastaa ensisijaisesti projektipäällikkö. Jokainen ryhmän jäsen on kuitenkin vastuussa hänelle osoitettuun tehtävään, tulokseen tai muuhun vastuualueeseen liittyvästä tiedotuksesta.

Projektipäällikkö toimittaa asiakkaalle vähintään kerran viikossa lyhyen **välikat-sauksen**, jossa hän kuvaa, mitä projektiryhmä on edellisen katsauksen tai kokouksen jälkeen tehnyt ja mitkä tekijät ovat vaikuttaneet projektin etenemiseen. Jos projektin eteneminen esimerkiksi viivästyi jonkin ongelman takia, välikatsauksessa raportoidaan siitä. Samalla kerrotaan, miten ongelma ratkaistiin tai miten se aiotaan ratkaista. Välikatsaus sisältää myös lyhyen kuvauksen siitä, mitä projektiryhmä aikoo tehdä seuraavaksi. Jos projekti on edennyt erityisen nopeasti tai ongelmia on ilmennyt poikkeuksellisen paljon, voidaan välikatsaus lähettää esimerkiksi joka toinen päivä.

Ryhmän jäsenten keskinäinen tiedotus hoidetaan pääosin silmäkkäin. Jäsenet pyrkivät työskentelemään keskenään samassa tilassa, jotta viestintä ryhmän jäsenten välillä olisi mahdollisimman tehokasta. Jos suullinen tiedotus ei ole mahdollista, hoidetaan yhteydenpito puhelimitse, sähköpostitse tai pikaviestimien välityksellä.

Ohjaajille ja tilaajan edustajille suunnattu tiedotus hoidetaan ensisijaisesti yhteisen sähköpostilistan kautta. Sähköpostilistan osoite on kuvatus@korppi.jyu.fi, ja sen jakelulistalle kuuluvat kaikki projektiorganisaation edustajat. Kaikki listalle lähetetyt viestit tallennetaan sähköpostiarkistoon, joka on nähtävillä osoitteessa <https://korppi.jyu.fi/kotka/servlet/list-archive/kuvatus/>.

Projektiryhmän jäsenten ja ohjaajien käytössä on myös sisäinen sähköpostilista kuvatus_opetus@korppi.jyu.fi. Opetuslistan sähköpostiarkisto sijaitsee osoit-

teessa https://korppi.jyu.fi/kotka/servlet/list-archive/kuvatus_opetus/.

5.3 Tiedostojen nimeäminen ja hakemistorakenne

Lähdekooditiedostojen nimeämisessä tullaan käyttämään Rubyn [6] ja JavaScriptin [2] yleisiä käytänteitä. Tiedostojen ja hakemistojen nimet kirjoitetaan englanniksi ja vain pieniä kirjaimia käyttäen. Välilyönnit tiedostonimissä korvataan alaviivalla. Lähdekooditiedostojen hakemistorakenne on valmiiksi määritelty Ruby on Rails -ohjelmointiympäristössä, eikä sitä muuteta.

Dokumenttiedostot nimetään sisältöä kuvaavilla nimillä ja dokumentin kielen mukaisesti, kuten `projektisuunnitelma.tex`. Ohjaajille tai asiakkaan edustajille julkaistuihin laajoihin pdf-dokumentteihin liitetään tiedostonimien perään versionumero (kts. luku 5.6) esimerkiksi muodossa `projektiraportti_v_[numero]_[numero]_[numero].pdf`.

Kokouksiin liittyvät dokumenttiedostot nimetään seuraavasti:

- Kokousten raakatekstimuotoiset esityslistat nimetään `esityslista_[järjestysnumero]_[päivämäärä].txt`.
- \TeX -muotoisten pöytäkirjojen nimeämistapa on `poytakirja_[järjestysnumero]_[päivämäärä].tex`.
- Pdf-muotoisten pöytäkirjojen nimeämistapa on `poytakirja_[järjestysnumero]_[päivämäärä].pdf`.
- Pdf-muotoiset tilakatsaukset nimetään `tilakatsaus_[järjestysnumero]_[päivämäärä].pdf`.

Päivämäärä esitetään muodossa `kkpp`. Esimerkiksi maaliskuun ensimmäinen päivä merkitään siis muodossa `0301`.

Projektin tulokset tallennetaan CD-levylle ja projektin WWW-hakemistoon päähakemiston alle seuraavan **hakemistorakenteen** mukaisesti:

```
ajankaytto
application_report
class documentation
esittelyt
itsearviointit
kayttoliittymademot
    html_luonnos_1
    html_luonnos_2
    js_luonnos_1
    js_luonnos_2
    tyokaluluonnokset
kokoukset
    esityslistat
    poytakirjat
    tilakatsaukset
ohjeet
raportit
    projektiraportti
    testausraportit
requirements specification
sahkopostiarkistot
    kuvatus
    kuvatus_opetus
sitoumus_ja_lisenssit
source_code
suunnitelmat
    projektisuunnitelma
    testaussuunnitelma
```

5.4 Lähdekoodi

Sovelluksen Ruby-lähdekoodi toteutetaan noudattaen Rubyn yleisiä käytänteitä [6] ja ennalta määrättyjä RDoc-käytänteitä. Sovelluksen JavaScript-lähdekoodin toteutuksessa mukaillaan JavaScriptin yleisiä käytänteitä [2].

Lähdekoodissa käytetyt luokat, muuttujat ja aliohjelmat nimetään käyttäen mahdollisimman kuvaavia englanninkielisiä nimiä. Myös lähdekoodin kommentit kirjoitetaan englanniksi, ja ne pyritään muotoilemaan riittävällä tarkkuudella. **Lähdekoodissa** olevat kommenttirivit aloitetaan Rubyn #-merkillä ja JavaScriptissä kahdella /-merkillä

Alla on esimerkki edellä mainittuja käytänteitä noudattaen toteutetusta JavaScript-lähdekoodista.

```
// Destructor for the object - removes the list element
// as well as the canvas.
// Also removes the remaining JS objects.
this.destroy = function() {
    this.post_delete();
    this.canvas.remove();
    // Some candy for the list item removal!
    this.list_element.fadeOut(300,
    function() {
        $(this).remove();
    });
    delete selection_table[this.name];
    expandList();
    delete this;
}
```

5.5 Testaus

Toteutetun sovelluksen verifiointi hoidetaan yksikkö-, integraatio- ja järjestelmätastauksella. Testauksella siis pyritään löytämään virheitä **lähdekoodista** ja varmistamaan siitä, että sovellukselle asetetut vaatimukset toteutuvat. Sovelluksen validoimiseksi pyritään saamaan tilaajan edustajilta mahdollisimman paljon käyttökoke-

muksia ja muita kommentteja. Tätä varten projektiryhmällä on käytössään testipalvelin, jossa tilaajan edustajilla on mahdollisuus kokeilla sovelluksen viimeisintä versiota testiversiota.

Yksikkötestauksesta vastaa ensisijaisesti koodin laatinut henkilö. Ohjelmoija testaa koodin kirjoittamisen yhteydessä, kun ohjelmoija on mielestään toteuttanut testattavan ominaisuuden.

Kun sovelluksen looginen osakokonaisuus valmistuu, suorittaa siitä päävastuussa ollut kehittäjä sille **integraatiotestausta**. Laajojen kokonaisuuksien tapauksessa testaukseen osallistuvat myös muut ryhmän jäsenet.

Järjestelmätestauksesta vastuussa oleva henkilö laatii testauksen suorittamiseksi testaussuunnitelman. Testaussuunnitelma sisältää kullakin testauskerralla suoritettavat testitapaukset vaihe vaiheelta. Kaikki projektiryhmän jäsenet osallistuvat järjestelmätestaukseen. Järjestelmätestauskerran suorittanut henkilö laatii siitä testausraportin. Testausraportissa kuvataan testauskerralla suoritettujen testitapausten tulokset sekä havaitut virheet ja puutteet.

Sovelluksen **käytettävyyteen** kiinnitetään huomiota kaikkien kehitysvaiheiden aikana. Käytettävyyttä ei erikseen testata, mutta mahdolliset kriittiset puutteet, virheet ja epäloogisuudet kirjataan ylös viimeistään esiteltäessä sovellusta ohjaajille ja asiakkaan edustajille.

5.6 Versiohallinta ja -numerointi

Tulosten versiohallintaan käytetään Git-versiohallintaohjelmistoa. Sovelluksen lähdekoodi sijoitetaan Git-pohjaiseen YouSource-julkaisujärjestelmään, jossa se on koajan myös asiakkaan saatavissa. Kaikki sovellukseen ja projektiin liittyvä dokumentaatio tullaan myös säilyttämään YouSource-järjestelmässä.

Julkistetuissa dokumenttien ja sovelluksen lähdekoodien versioissa tullaan käyttämään **kolmiportaista versionumerointia**. Ryhmän sisäiset versiot aloitetaan versionumerosta 0.0.1 ja kunkin uuden version osalta kasvatetaan vähiten merkitsevää numeroa yhdellä. Tällöin toinen versio on versionumeroltaan 0.0.2. Projektiorganisaatiolle julkistettava versioiden numerointi aloitetaan versionumerosta 0.1.0. Seuraavat versiot numeroidaan kasvattamalla toisen tason numeroa yhdellä. Ensimmäisen hyväksytyyn version numero on 1.0.0 ja sitä seuraavissa asiakkaan hyväksy-

missä versioissa kasvatetaan toisen tason numeroa yhdellä (siis toinen hyväksytty versio on 1.1.0).

5.7 Katselmoinnit ja tulosten hyväksyminen

Lähdekoodi tullaan **katselmoimaan** vähintään kaksi kertaa projektin aikana. Katselmointiin osallistuu projektiryhmän lisäksi vähintään tekninen ohjaaja, joka kommentoi projektiryhmän kirjoittamaa lähdekoodia. Dokumentit katselmoidaan tarvittaessa projektipalaverissa.

Tulokset hyväksytetään vähintään projektin ohjaajilla ja tilaajan edustajista Tero Tuovisella tai hänen nimeämällään edustajalla. Tilaajan edustajan hyväksyntä vaaditaan vähintään sovellukselle, käyttöohjeelle ja sovellusraportille. Sovelluksen lähdekoodin hyväksyy tekninen ohjaaja. Vastaava ohjaaja hyväksyy tuloksista projektisuunnitelman ja -raportin sekä sovellusraportin.

5.8 Tulosten koostaminen ja toimittaminen

Projektiryhmä kokoaa projektin tulokset **projektikansioon** ja CD-levylle. Tilaajalle toimitetaan projektin tulokset vähintään yhdellä **CD-levyllä**. Kopiot levyistä tallennetaan myös projektikansioon ja tietotekniikan laitoksen arkistoon. Myös kukin ryhmän jäsen saa halutessaan tulokset CD-levyllä. Projektiryhmä sijoittaa kaikki tulokset myös YouSource-sivustolle.

Projektiryhmä koostaa ja luovuttaa CD-levyt vasta, kun kaikki tulokset on hyväksytty. YouSource-järjestelmään tuloksia toimitetaan sitä mukaa kuin ne valmistuvat.

6 Tehtävät, työmäärät ja vastualueet

Luvussa määritellään ryhmän projektipäällikön ja varapäällikön vastualueet, sekä esitellään jäsenten muut vastualueet. Lisäksi määritellään eri tehtäväkokonaisuuksille varatut työmäärät ja ryhmän sisäinen työnjako.

6.1 Projektipäällikkö ja varapäällikkö

Projektipäällikkönä toimii Jussi Perttola ja varapäällikkönä Pekka Iso-Ahola. Projektipäällikön vastuutehtäviin kuuluvat projektin suunnittelu ja hallinta, ajankäytön ja projektin etenemisen seuranta, tiedotus sekä ryhmän sisäisten tehtävien jakaminen. Projektipäällikkö on vastuussa myös projektisuunnitelman ja -raportin laatimisesta. Varapäällikkö toimii projektipäällikön sijaisena tämän poissaollessa.

6.2 Vastualueet tulosten osalta

Keskeisimpien tulosten vastuuhenkilöt on esitetty taulukossa 6.1. Vastuuhenkilö ei yksinään toteuta tulosta, mutta vastaa sen valmistumisesta, tarkastettavaksi toimitamisesta ja mahdollisesti vaadituista muokkauksista. Myös tiedottaminen kyseisen tuloksen valmistumisesta kuuluu siitä vastaavan henkilön tehtäviin.

Tulos	Vastuuhenkilö
Projektisuunnitelma	Jussi Perttola
Vaatimusmäärittely	Tommi Tuovinen
Järjestelmätestaus	Tommi Tuovinen
Lähdekoodin viimeistely	Pekka-Iso-Ahola
Sovellusraportti	Pekka Iso-Ahola
Projektiraportti	Jussi Perttola

Taulukko 6.1: Vastualueet keskeisimpien tulosten osalta.

Toteutettavan sovelluksen osien vastualueet jaetaan ryhmän jäsenten kesken. Lähdekoodista, siihen liittyvistä tuloksista ja niiden toimittamisesta projektiorganisaatiolle vastaavat ryhmän kaikki jäsenet yhteisvastuullisesti.

6.3 Tehtäväkohtaiset työmäärät ja työnjako

Projektiryhmän jäsenet ovat sitoutuneet työskentelemään projektin eteen noin 30–40 tuntia viikoittain oheiskursseihin kuuluvien tuntien lisäksi. Tämä tarkoittaa, että ryhmän jäsenet saavuttavat henkilökohtaisen työmäärätavoitteensa 280 tuntia 7–10 viikon aikana.

Projektiryhmän jäsenten sovellusprojektiin ja siihen liittyviin oheiskursseihin käytämät työtunnit on arvioitu taulukossa 6.2 tehtäväkokonaisuuksittain ja tehtävittäin. Työtuntiarvio perustuu sovelluksen eri osakokonaisuuksien arvioituihin vaativuuksiin. Arviot on pyritty pitämään mahdollisimman tarkkoina, eikä niihin ole lisätty erillistä virhemarginaalia. Tästä syystä on todennäköistä, että joillekin tehtäville arvioidut työmäärät tulevat ylittymään. Jos tuntiarviot ylittyvät huomattavasti, tulee arvioida, joudutaanko joitain suunniteltuja ominaisuuksia rajaamaan pois.

Tehtäväkokonaisuus	Tehtävä	PIA	JP	TT	Kaikki
Projektin hallinta					
	Suunnittelu	0	30	0	30
	Projektisuunnitelma	0	40	0	40
	Seuranta	0	15	0	15
	Tiedotus	0	12	0	12
	Muut tehtävät	2	5	5	12
	Projektiraportti	0	25	0	25
	Viimeistely	4	6	4	14
	Tulosten luovutus	4	6	4	14
	Yhteensä	10	139	13	162
Kokoukset					
	Valmistelu	2	4	2	8
	Kokoukset	20	20	20	60
	Pöytäkirjat	10	7	12	29
	Yhteensä	32	31	34	97
Esitutkimus					
	Aihealueeseen tutustuminen	7	6	7	20
	Koulutus	6	6	6	18
	Työkaluihin tutustuminen	12	10	15	37
	Yhteensä	25	22	28	75
Vaativuusmäärittely					
	Suunnittelu	0	3	10	13
	Raportointi	0	1	20	21
	Yhteensä	0	4	30	34
Suunnittelu					
	Sovelluksen rakenne	3	3	3	9
	Kuvien avaaminen	5	4	5	14
	YouDataan integrointi	5	4	6	15
	Opetusaineiston muodostaminen ja tallennus	6	6	8	20
	Rajapinnat	6	4	6	16
	Valintojen tallennus	8	6	6	20
	Valintatyökalut	8	6	4	18
	WWW-käyttöliittymä	9	8	10	27
	Yhteensä	50	41	48	139
Toteutus					
	Kuvien avaaminen	15	2	3	20
	YouDataan integrointi	4	4	23	31
	Opetusaineiston muodostaminen ja tallennus	20	4	4	28
	Rajapinnat	4	2	23	29
	Valintojen tallennus	25	4	5	34
	Valintatyökalut	25	4	20	49
	WWW-käyttöliittymä	25	6	20	51
	Yhteensä	118	26	98	242
Testaus					
	Suunnittelu	1	1	8	10
	Järjestelmätestaus	2	2	6	10
	Raportointi	0	0	4	4
	Yhteensä	3	3	18	24
Viimeistely					
	Sovellusraportti	28	3	0	31
	Katselmointi	4	4	4	12
	Lähdekoodin viimeistely	6	3	3	12
	Sovelluksen luovutus	4	4	4	12
	Yhteensä	42	14	11	67
	Projektin tunnit yhteensä	280	280	280	840
Oheiskurssit					
	Kirjoitusviestintä	25	25	25	75
	Puheviestintä	15	15	15	45
	Projektiluennot	15	15	15	45
	Yhteensä	55	55	55	165
	Kaikki tunnit yhteensä	335	335	335	1005

Taulukko 6.2: Tehtävien arvioidut työtunnit.

7 Prosessimalli ja aikataulu

Luvussa kuvataan projektissa käytettävää prosessimallia ja määritellään projektin aikataulu.

7.1 Prosessimalli

Kuvatus-projektissa käytetään ketterien menetelmien arvojen mukaista prosessimallia, jossa sovellusta kehitetään inkrementaalisesti ja iteratiivisesti läpi sen elinkaaren. Projektissa ei käytetä mitään tiettyä menetelmää tai prosessimallia, vaan käytettävä malli on räätälöity toteutettavaa projektia varten. Prosessimalli on pääosin inkrementaalinen, sillä uusia ohjelmakomponentteja otetaan toteutettavaksi koko sovelluksen kehityksen ajan.

Jokaisessa projektikokouksessa esitellään sen hetkinen versio sovelluksesta ja keskustellaan tilaajan edustajien kanssa sen toiminnallisuuksista. Samassa yhteydessä kerrotaan, miten kyseiseen versioon suunnitellut ominaisuudet on toteutettu. Samalla päivitetään tarvittaessa toiminnallisuuksien prioriteettejä ja sovitaan, mitkä niistä pyritään toteuttamaan seuraavaan kokoukseen mennessä.

7.2 Aikataulu

Projekti on suunniteltu toteutettavaksi vuoden 2012 helmikuun alun ja huhtikuun lopun välillä. **Helmikuun** aikana projektiryhmä tutustuu aiheeseen ja työkaluihin sekä määrittelee ja priorisoi tietojärjestelmän vaatimukset. Samalla projektipäällikkö suunnittelee projektin läpiviennin. Sovelluksen varsinainen kehitys alkaa maaliskuun alussa ja päättyy ennen huhtikuun puoltaväliä. Projektin viimeistelyyn varataan **huhtikuun loppupuoli**, jonka aikana sovellus testataan sekä laaditaan projekti- ja sovellusraportti. Huhtikuun loppuun mennessä projektin tulokset koostetaan ja luovutetaan asiakkaalle.

Projektin suunniteltu kesto esitetään vaiheineen kuvassa 7.1. Kyseinen Gantt-kaavio ei esitä toteutuksen vaiheita, mutta ne ovat luettavissa tehtäväkokonaisuuksien aikataulullisesta jaksotuksesta. Taulukossa 6.2 esitetyt työtuntiarviot on otettu huomioon Gantt-kaavion suunnittelussa.



Kuva 7.1: Projektin suunniteltu aikataulu.

8 Riskien hallinta

Luvussa arvioidaan projektiin liittyvien riskien todennäköisyyksiä ja haittavaikutuksia. Samalla määritellään keinoja ehkäistä ja ennakoida riskejä. Lisäksi kuvataan miten riskien toteutuessa konkreettisesti toimitaan.

8.1 Arvioidut riskit, niiden todennäköisyydet ja haittavaikutukset

Riskien esiintymistodennäköisyyksiä ja niiden haittavaikutuksia arvioidaan kolmiportaisella asteikolla *pieni*, *keskinkertainen* ja *suuri*. Projektiin liittyvät riskit on järjestetty taulukossa 8.1 kahteen ryhmään. Ensimmäisenä esitetään projektin eri osa-alueisiin liittyvät riskit ja sen jälkeen jäsenten toimintaan liittyvät riskit. Näiden ryhmien sisällä olevat riskit on järjestetty niiden esiintymistodennäköisyyksien ja haittavaikutusten suuruuden perusteella.

Riski	Todennäköisyys	Haittavaikutus
Sidosryhmien toiminnan viiveet	suuri	keskinkertainen
Puutteet projektiryhmän tietotaidoissa	keskinkertainen	keskinkertainen
Muutostarpeet projektiryhmän tietokoneissa	keskinkertainen	pieni
Ryhmän jäsenten odottamattomat poissaolot	pieni	suuri
Projektin hallinnan puutteet	pieni	keskinkertainen
Valitun alustan asettamat rajoitteet	pieni	pieni

Taulukko 8.1: Riskien arvioidut todennäköisyydet ja haittavaikutukset.

Luvuissa 8.2 – 8.7 kuvataan kunkin riskin hallintaa. Luvussa arvioidaan yksittäisten riskien vaikutuksia tuloksiin tai projektin läpivientiin, sekä miten riskin toteutuessa tulee toimia. Lisäksi esitetään tapoja, joilla yksittäisiä riskejä voidaan ennaltaehkäistä ja ennakoida.

8.2 Sidosryhmien toiminnan viiveet

Sidosryhmien toiminnan viiveillä tarkoitetaan projektiorganisaatioon tai muuhun sidosryhmään kuuluvien henkilöiden toimenpiteiden viivästymistä. Jos esimerkiksi joudutaan odottamaan testipalvelimen asentamista, liikutaan tämän riskin alueella.

Myös jos esimerkiksi loppuesittelyä joudutaan siirtämään myöhemmäksi asiakkaan edustajien tai vastaavan ohjaajan muiden kiireiden takia, on riski toteutunut. Toteutuessaan riski hidastaa projektin etenemistä ja hankaloittaa aikataulussa pysymistä.

Riskin ehkäisemiseksi projektiryhmän jäsenet pitävät aktiivisesti yhteyttä sidosryhmien kanssa ja sopivat toimenpiteistä mahdollisimman varhain, ettei kiirettä pääse syntymään. Myös riskin ennakointi perustuu jatkuvaan molemminpuoliseen yhteydenpitoon projektiryhmän jäsenten ja sidosryhmien välillä.

Riskin toteutuessa arvioidaan ensin, koskeeko se projektiorganisaation ulkopuolisia sidosryhmiä. Jos näin on, arvioidaan asiakkaan kanssa riskin aiheuttamat haittavaikutukset ja muutetaan sen perusteella esimerkiksi sovelluksen vaatimuksia tai niiden priorisointeja. Jos toteutunut riski johtuu projektiorganisaatioon kuuluvista henkilöistä, pyritään asiaan löytämään ratkaisu neuvottelemalla aikatauluista projektiorganisaation jäsenten kesken.

8.3 Puutteet projektiryhmän tietotaidoissa

Jokaisella projektiryhmän jäsenellä on omat vahvuutensa ja heikkoutensa. Ryhmän tietotaidot eivät kokonaan vastaa projektissa vaadittavia tietotaitoja, joten eteen saattaa tulla tilanteita, joita ryhmän jäsenet eivät osaa aiempien tietojensa perusteella ratkaista. Tällainen tilanne voi koskea esimerkiksi ohjelmointiratkaisuja, mutta se voi myös edellyttää aihealueen perusteellista tuntemista. Riskin toteutuessa projektiryhmä joutuu käyttämään työaikaansa ongelmien selvittämiseen, jolloin projektin eteneminen hidastuu ja saatetaan jopa joutua luopumaan suunnitelluista toiminnallisuuksista.

Riskin ehkäisemiseksi on pyritty varaamaan riittävästi aikaa aihealueeseen ja työkaluihin tutustumiseen. Ryhmälle on osoitettu tekninen ja vastaava ohjaaja, jotka voivat auttaa ongelmien ratkaisemisessa, mutta myös tilaajan edustajilta voi pyytää neuvoa tarvittaessa. Riskin ennakkoinnissa auttavat viikoittaiset kokoukset ja tiivis yhteydenpito projektiorganisaation sisällä. Hyvä yhteydenpito takaa, että ongelmaan osataan puuttua mahdollisimman ajoissa.

Riskin toteutuessa projektiryhmän tulee ottaa yhteyttä ohjaajiin ja tilaajan edustajiin projektiorganisaation sähköpostilistan kautta. Jos asia ei ratkea, tulee tehtävästä keskustella asiakkaan kanssa perusteellisemmin ja arvioida, voidaanko tehtävä rajata projektin ulkopuolelle tai siirtää projektin myöhempään kehitysvaiheeseen.

8.4 Muutostarpeet projektiryhmän tietokoneissa

Projektiryhmän tietokoneissa tarvitaan projektin aikana useita eri ohjelmia ja niiden eri versioita. Projektin alussa suoritetusta tarpeellisten ohjelmien kartoituksesta huolimatta on odotettavissa, että tarve uusille ohjelmille tai niiden lisäosille ilmenee projektin aikana.

Koska projektiryhmän jäsenillä ei ole ylläpito-oikeuksia Windows-kehityskoneelle, on ryhmä riippuvainen ATK-lähituen palveluista. Riskinä on, että ryhmä joutuu odottamaan uusien ohjelmien asennusta pitkään tai että ohjelmaa ei ole mahdollista saada lainkaan. Riski liittyy siis myös luvussa 8.2 esitettyyn riskiin.

Toteutuessaan riski aiheuttaisi sen, ettei ryhmä voi käyttää haluamaansa ohjelmaa projektiryhmän tietokoneilla. Tämän ehkäisemiseksi kehitystyössä tulee käyttää yleisesti tunnettuja ja hyväksi havaittuja ohjelmia mahdollisimman paljon. Riskiä ei voi ennakoida. Se tulee esille silloin, kun huomataan jokin tarve kehityskoneen muutokseen.

Ongelman ilmetessä ryhmän tulee välittömästi ottaa yhteyttä ATK-lähitukeen ja selvittää tarvittavien ohjelmistojen saatavuus. Jos tarvittavaa ohjelmaa ei ole saatavissa, tulee valita käyttöön jokin vaihtoehtoinen ohjelmisto.

8.5 Jäsenten odottamattomat poissaolot

Projektiryhmän jäsenet ovat sitoutuneet tiiviiseen projektityöskentelyyn, ja projektin suorittaminen edellyttää jokaisen työpanosta. Jos joku ryhmän jäsenistä joutuu odottamattomasti olemaan poissa, saattaa se viivästyttää projektin etenemistä huomattavasti.

Riskin ehkäisemiseksi projektin tehtävät jaetaan siten, etteivät projektin keskeisimmät tehtävät ole täysin kenenkään yksittäisen ryhmän jäsenen vastuulla. Tarvittaessa kenen tahansa jäsenen tulee pystyä jatkamaan jonkun toisen aloittamaa työtä. Odottamattomat poissaolot ovat odottamattomia, eikä niitä voi ennakoida.

Toteutuessaan riski edellyttää tehtävien uudelleenorganisointia ryhmän sisällä. Pitkä poissaolo voi vaikuttaa projektin etenemiseen huomattavasti. Tällaisessa tilanteessa projektiryhmän tulee neuvotella asiakkaan kanssa jatkotoimenpiteistä. Projektipäällikön odottamattomiin poissaoloihin on varauduttu nimeämällä varapäällikkö, jonka tehtävät kuvataan luvussa 6.1.

8.6 Projektin hallinnan puutteet

Projektiryhmän jäsenillä ei ole aiempaa kokemusta sovellusprojektin suunnittelusta ja hallinnasta. Puutteellinen projektin hallinta saattaa johtaa esimerkiksi epäselvyyksiin työnjaossa tai ryhmähengen heikkenemiseen. Riskin toteutumistodennäköisyys on pieni, sillä ryhmän jäsenet ovat tiiviisti sitoutuneet projektin suorittamiseen.

Ehkäisevänä toimenpiteenä on jäsenten keskinäinen yhteydenpito ja tehtävämäärien kirjallinen arviointi. Lisäksi ryhmän vastaava ohjaaja valvoo ryhmän toimintaa ja tarvittaessa puuttuu siihen.

Projektipäällikkö raportoi ryhmän tekemisistä projektiorganisaatiolle, mikä pakottaa ryhmän tarkastelemaan projektin tilannetta päivittäin. Jos havaitaan, että ryhmän työnjako on epäselvä tai työilmapiiri on huono, voidaan ennakoida riskin toteutumisen olevan käsillä.

Riskin toteutuessa ongelmatilanteeseen puututaan ja otetaan tarvittaessa yhteyttä vastaavaan ohjaajaan. Projektiryhmän jäsenten käyttämät työtunnit ja projektisuunnitelma tarkastetaan ongelmakohtien paikallistamiseksi. Ääritilanteessa ryhmälle voidaan valita uusi projektipäällikkö.

8.7 Valitun alustan asettamat rajoitteet

Riski liittyy olennaisesti Ruby on Rails -sovelluskehikseen, joka ei ole ryhmän jäsenille ennestään tuttu. Tästä saattaa aiheutua ongelmia sovelluksen kehitysvaiheessa. Riski on kuitenkin arvioitu sekä esiintymistodennäköisyydeltään, että haittavaikutuksiltaan pieneksi, sillä sovellukseen integroitava YouData on myös toteutettu käyttäen Ruby on Railsia, minkä pitäisi vähentää yhteensopivuusongelmia.

Riskin ehkäisemiseksi ryhmän jäsenet opiskelevat projektin alkuvaiheessa ja koko ajan kehitystyön rinnalla Ruby on Rails -ohjelmointia. Pienemmissä ongelmatilanteissa apua löytyy ohjelmointioppaista, tekniseltä ohjaajalta ja tilaajan edustajista Marko Peltolalta.

Riskin ennakointi on vaikeaa, sillä ilman aiempaa kokemusta on ryhmän jäsenten hankala arvioida, voidaanko jotain ominaisuutta toteuttaa käytössä olevilla välineillä. Ennakointia voidaan tehdä teknisen ohjaajan kanssa, jolla on enemmän kokemusta valitusta alustasta.

Jos riski toteutuu, eikä jotain ominaisuutta voida toteuttaa valitulla kielellä, pyydetään tekniseltä ohjaajalta ratkaisu tilanteeseen. Tarvittaessa yksittäisiä ominaisuuksia esimerkiksi voidaan toteuttaa jollain toisella ohjelmointikielellä.

9 Yhteenveto

Kuvatus-projekti toteuttaa Jyväskylän yliopiston tietotekniikan laitokselle toiminnallisen prototyypin sovelluksesta kuva-analyysimenetelmissä käytettävän opetusdatan valitsemiseen ja hallintaan. Kehitettävällä sovelluksella määritetään oppivien tietokoneohjelmien koulutusaineistoja rajaamalla alueita hyperspektrikuvista ja määrittelemällä rajauksille luokkia. Valmis sovellus helpottaa tutkijoiden työtä ja yhdenäistää tutkimusryhmien käytänteitä.

Projektin keskeisin haaste on tehdä sovelluksesta niin laadukas, että se soveltuu tutkijoiden päivittäiseksi työkaluksi. Myös eri ominaisuuksien toteutuksen priorisointi korostuu, koska sovellusalue on niin laaja, ettei kaikkia ominaisuuksia ehditä toteuttaa sovellusprojektin aikana.

Projekti alkoi tammikuussa 2012 ja toimiva prototyyppi luovutetaan asiakkaalle huhti-toukokuun vaihteessa. Projektiryhmän jäsenet oppivat kurssin aikana projektin suunnittelussa ja hallinnassa vaadittavia taitoja, ryhmätyöskentelyä, kokouskäytänteitä sekä projektiviestintää. Lisäksi opiskelijat oppivat ohjelmistokehityksessä vaadittavia käytännön taitoja.

10 Lähteet

- [1] Kent Beck, Mike Beedle, Alistair Cockburn, et al., "Agile Manifesto", saatavilla HTML-muodossa <URL: <http://agilemanifesto.org/>>, viitattu 2.3.2012.
- [2] Douglas Crockford, "Code Conventions for the JavaScript Programming Language", saatavilla HTML-muodossa <URL: <http://javascript.crockford.com/code.html>>, viitattu 8.3.2012.
- [3] Petri Heinonen, Ajankäytönseurantasovellus, saatavilla Excel-muodossa <URL: <http://appro.mit.jyu.fi/tools/ajankaytto/ajankaytonseuranta.xls>>, Jyväskylän yliopisto, informaatioteknologian tiedekunta, viitattu 23.2.2011.
- [4] Sam Ruby, Dave Thomas, David Heinemeir Hansson, et al. "Agile Web Development with Rails", Pragmatic Bookshelf, 2010.
- [5] Ruby community, "Ruby Programming Language", saatavilla HTML-muodossa <URL: <http://www.ruby-lang.org/en/>>, viitattu 28.2.2012.
- [6] Savvica Inc., "Ruby Coding Convention", saatavilla HTML-muodossa <URL: <http://rails.nuvvo.com/lesson/5017-ruby-coding-convention>>, viitattu 2.3.2012.
- [7] Olli Wirpi, "Judo-sovellusprojekti, Projektisuunnitelma", Jyväskylän yliopisto, tietotekniikan laitos, 31.3.2011.