

Hälyri-Sovellusprojekti

**Niko Mononen
Veli-Mikko Puupponen
Ilkka Rautiainen
Atte Söderlund**

Projektiraportti

Versio: 1.0.0
Julkinen
27. kesäkuuta 2014

Jyväskylän yliopisto

Tietotekniikan laitos

Jyväskylä

Hyväksyjä	Päivämäärä	Allekirjoitus	Nimenselvennys
Projektipäällikkö	__.__.2014		
Tilaaaja	__.__.2014		
Ohjaaja	__.__.2014		

Tietoa dokumentista

Tekijät:

- | | |
|------------------------------|--|
| • Niko Mononen (NM) | <code>niko.m.mononen@student.jyu.fi</code> |
| • Veli-Mikko Puupponen (VMP) | <code>veli-mikko.a.puupponen@student.jyu.fi</code> |
| • Ilkka Rautiainen (IR) | <code>ilkka.t.rautiainen@student.jyu.fi</code> |
| • Atte Söderlund (AS) | <code>atte.a.soderlund@student.jyu.fi</code> |

Dokumentin nimi: Hälyri-projekti, Projektiraportti

Sivumäärä: 55

Tiedosto: `Halyri_projektiraportti_1.0.0.tex`

Tiivistelmä: Hälyri-projekti suoritti alustavan toteutettavuuskartoituksen ja kehitti hätäilmoitusjärjestelmän prototyypin Sovellusprojekti-kurssilla keväällä 2014. Projektiraportissa kuvataan projektin suunniteltua ja toteutunutta läpivientiä käsitellen muun muassa tavoitteita, resursseja ja käytänteitä sekä jäsenten oppimaa ja kokemuksia. Lisäksi projektiraportissa verrataan projektin toteutumaa suunniteltuun mm. työnjaon ja työmäärien, aikataulun sekä arvioitujen riskien hallinnan osalta ottaen kantaa erojen syihin ja vaikutuksiin.

Avainsanat: Aikataulu, EKG, haasteet, hätäkeskus, hätäpuhelu, kokemuksia, käytänteet, matkapuhelin, ohjelmistoprojekti, opittua, projektiraportti, prosessi, resurssit, riskien hallinta, sovellus, tavoitteet, taustaa, tehtävät, työmäärät, vastuualueet, videopuhelu, Windows Phone 8.

Muutoshistoria

Versio	Päivämäärä	Muutokset	Tekijät
0.0.1	6.5.2014	Projektiraportin laatiminen on aloitettu.	NM
0.0.2	10.5.2014	Kirjoitettiin projektin taustasta, tavoitteista ja projektioorganisaatiosta.	NM
0.0.3	18.5.2014	Kirjoitettiin käytänteistä, työmääristä, työnjaosta ja riskeistä.	NM
0.0.4	22.5.2014	Kirjoitettiin tietojärjestelmälle asetettujen tavoitteiden toteutumisesta.	NM
0.1.0	24.5.2014	On korjattu kirjoitusvirheitä.	NM
0.1.1	27.5.2014	Raporttia korjattiin saadun palautteen mukaisesti.	NM
0.2.0	2.6.2014	Analysoitiin tavoitteiden saavuttamista.	NM
0.2.1	8.6.2014	Raporttia on korjattu saadun palautteen mukaisesti.	NM
0.2.2	17.6.2014	Lisättiin toteutunut aikataulu.	NM
0.2.3	17.6.2014	Analysoitiin toteutunutta aikataulua ja työtunteja.	NM
0.2.4	21.6.2014	Analysoitiin käytettyjä tunteja.	NM
0.3.0	22.6.2014	Kirjoitettiin yhteenvetoja ja lukujen johdantokappaleita.	NM
0.4.0	24.6.2014	Raporttia korjattiin saadun palautteen mukaisesti.	NM
0.5.0	26.6.2014	Raporttia korjattiin saadun palautteen mukaisesti.	NM
1.0.0	27.6.2014	Raporttia korjattiin saadun palautteen mukaisesti.	NM

Tietoa projektista

Projekti kehitti prototyypin uudentlaisesta tietojärjestelmästä, joka mahdollistaa yhteyden hädässä olevien ihmisten ja hätäkeskusten välillä. Keskeinen ajatus on hyödyntää matkapuhelinten merkittävästi lisääntynttä datan lähetys- ja vastaanotto-kykyä osana auttamisprosessia. Tavoitteena kehitetyllä tietojärjestelmällä on luoda aikaisempaa tarkempi kuva hätätilanteesta entistä lyhyemmässä ajassa. Tämä auttaa pelastamaan ihmishenkiä, parantamaan palvelun laatua ja tehostamaan resursien käyttöä. Projektissa kehitetyn hätäilmoitusjärjestelmän prototyypin avulla osoitettiin teknologiset mahdollisuudet ja haasteet sekä havainnollistettiin käyttöliittymillä avuntarvitsijan ja hätäkeskuksen vuorovaikutusta.

Tekijät:

- | | |
|------------------------------|--|
| • Niko Mononen (NM) | <code>niko.m.mononen@student.jyu.fi</code> |
| • Veli-Mikko Puupponen (VMP) | <code>veli-mikko.a.puupponen@student.jyu.fi</code> |
| • Ilkka Rautiainen (IR) | <code>ilkka.t.rautiainen@student.jyu.fi</code> |
| • Atte Söderlund (AS) | <code>atte.a.soderlund@student.jyu.fi</code> |

Tilaaaja:

- | | |
|--------------------|--|
| • Vesa Lappalainen | <code>vesa.t.lappalainen@jyu.fi</code> |
| • Tero Tuovinen | <code>tero.tuovinen@jyu.fi</code> |

Ohjaajat:

- | | |
|------------------------|--|
| • Jaakko Kosonen | <code>jaakko.j.kosonen@gmail.com</code> |
| • Olli Kauppinen | <code>olli.p.kauppinen@student.jyu.fi</code> |
| • Jukka-Pekka Santanen | <code>santanen@mit.jyu.fi</code> |

Asiantuntijat:

- | | |
|-----------------|-----------------------------------|
| • Helena Jantti | <code>helena.jantti@kuh.fi</code> |
|-----------------|-----------------------------------|

Yhteystiedot:

- Sähköpostilistat: `halyri@korppi.jyu.fi,`
`halyri_opetus@korppi.jyu.fi,`
`appl12.group@korppi.jyu.fi.`
- Sähköpostiarkistot: `https://korppi.jyu.fi/kotka/`
`servlet/list-archive/halyri/,`
`https://korppi.jyu.fi/kotka/`
`servlet/list-archive/halyri_opetus/,`
`https://korppi.jyu.fi/kotka/`
`servlet/list-archive/appl12.group/`

Sisältö

1	Johdanto	1
2	Termit	2
3	Taustaa ja tavoitteiden toteutuminen	4
3.1	Taustaa ja tarpeita	4
3.2	Prototyypille asetettujen tavoitteiden toteutuminen	5
3.3	Oppimistavoitteiden toteutuminen	8
3.4	Tulokset ja niiden toteutuminen	8
4	Organisaatio ja resurssit	10
4.1	Projektiorganisaatio	10
4.2	Projektin tilat ja laitteet	11
4.3	Dokumentointityökalut	12
4.4	Ohjelmointityökalut	12
4.5	Luennot ja perehdytykset	13
5	Projektin käytänteet	14
5.1	Tiedotus	14
5.2	Palaverit	15
5.3	Tiedostojen nimeäminen	16
5.4	Hakemistorakenne	16
5.5	Lähdekoodi	17
5.6	Versiointi ja versiohallinta	20
5.7	Testaus	20
5.8	Tulosten katselmoinnit ja hyväksyminen	21
5.9	Tulosten koostaminen ja toimittaminen	22
6	Tehtävät, työmäärät ja työnjako	23
6.1	Vastuualueet	23
6.2	Työnositus ja työmäärä	24
6.3	Ryhmän työtunnit tehtäväkokonaisuuksittain	29
6.4	Niko Monosen työtunnit tehtäväkokonaisuuksittain	30
6.5	Veli-Mikko Puupposen työtunnit tehtäväkokonaisuuksittain	31
6.6	Ilkka Rautiaisen työtunnit tehtäväkokonaisuuksittain	32
6.7	Atte Söderlundin työtunnit tehtäväkokonaisuuksittain	33

7 Ohjelmistokehitysprosessi ja aikataulu	34
7.1 Prosessi	34
7.2 Aikataulu	35
7.3 Ryhmän työtunnit viikoittain	38
7.4 Niko Monosen työtunnit viikoittain	39
7.5 Veli-Mikko Puupposen työtunnit viikoittain	40
7.6 Ilkka Rautiaisen työtunnit viikoittain	41
7.7 Atter Söderlundin työtunnit viikoittain	42
8 Riskit ja niiden hallinta	43
8.1 Riskien todennäköisyydet ja haitat	43
8.2 Alustan rajoitukset	44
8.3 Kehittäjien tietotaidon puutteet	44
8.4 Tavoitteiden rajaaminen	45
8.5 Tiedotuksen puutteet	46
8.6 Jäsenten poissaolot	46
8.7 Kokemattomuus projektihallinnassa	47
9 Jäsenten kokemuksia ja oppimaa	48
9.1 Niko Monosen kokemuksia ja oppimaa	48
9.2 Veli-Mikko Puupposen kokemuksia ja oppimaa	49
9.3 Ilkka Rautiaisen kokemuksia ja oppimaa	50
9.4 Atte Söderlundin kokemuksia ja oppimaa	51
10 Yhteenveto	52
11 Lähteet	53

1 Johdanto

Projekti suoritti Sovellusprojekti-kurssilla keväällä 2014 alustavan toteutettavuuskartoituksen ja kehitti prototyypin uudenlaisesta hätäilmoitusjärjestelmästä, joka mahdollistaa yhteyden hädässä olevien ihmisten ja hätäkeskusten välillä. Keskeinen ajatus on hyödyntää matkapuhelinten merkittävästi lisääntynyttä datan lähetys- ja vastaanottokykyä osana auttamisprosessia. Tavoitteena kehitetyllä tietojärjestelmällä on luoda aikaisempaa tarkempi kuva hätätilanteesta entistä lyhyemmässä ajassa. Tämä auttaa pelastamaan ihmishenkiä, parantamaan palvelun laatua ja tehostamaan resurssien käyttöä. Projektissa kehitetyn hätäilmoitusjärjestelmän prototyypin avulla osoitettiin teknologiset mahdollisuudet ja haasteet sekä havainnollistettiin käyttöliittymällä avuntarvitsijan ja hätäkeskuksen vuorovaikutusta.

Projektiraportissa kuvataan projektin toteutunutta läpivientiä esitellen sen tulokset, resurssit, käytänteet, prosessimallia, aikataulua ja riskien hallintaa. Lisäksi käsitellään projektiryhmän jäsenten tehtäviä, työmääriä ja tehtäväjakoa projektissa. Projektiraportin laatimisessa on hyödynnetty Paatti-projektin projektiraporttia [2] ja Sovellusprojektien ohjetta [18]. Projektissa laaditussa vaatimusmäärittelyssä [9] on kuvattu kehitetylle sovellukselle asetettuja vaatimuksia ja niiden toteumaa. Sovellusraportissa [12] on kuvattu sovelluksen käyttöliittymää, toteutusratkaisuja, havaittuja puutteita ja virheitä sekä jatkokehitysideoita. Projektin muut tulokset on esitelty luvussa 3.4.

Luvussa 2 kuvataan dokumentissa käytetyt termit. Luvussa 3 kuvataan projektin taustoja sekä tavoitteiden, tulosten ja projektiryhmän jäsenten oppimistavoitteiden toteutumista. Luvussa 4 esitellään projektiorganisaatio ja resurssit. Luvussa 5 kuvataan projektin käytänteet. Luvussa 6 tarkastellaan projektiryhmän tehtäviä sekä tehtävien työmääriä ja työnjakoa. Luvussa 7 kuvataan projektin ohjelmistokehitysprosessia ja aikataulua. Luvussa 8 kuvataan projektin ennakoitujen riskien hallintaa sekä käsitellään niiden syitä ja vaikutusta projektin läpivientiin ja tuloksiin. Luvussa 9 esitellään ryhmän jäsenten kokemuksia ja oppimaa projektista.

2 Termit

Dokumentin aihealueen termejä ovat seuraavat:

EKG	on elektrokardiogrammi eli sydänsähkökäyrä.
Hätäkeskus	vastaanottaa ja käsittelee yleiseurooppalaiseen hätänumeroon 112 soitetut hätäpuhelut.
Käyttöliittymä	on se ohjelmiston, laitteen tai minkä tahansa muun tuotteen osa, jonka kautta käyttäjä käyttää tuotetta.
Mobiilisovellus	on sovellus, joka on suunniteltu mobiilipäätelaitteisiin eli matkapuhelimiin, älypuhelimiin, kämmentietokoneisiin ja ammattikäyttöön tarkoitettuihin mobiilipäätelaitteisiin.
Palvelin	on palvelinsovellusta suorittava tietokone.
Palvelinsovellus	on ohjelma, joka tarjoaa erilaisia palveluja muille ohjelmille joko tietokoneverkon välityksellä tai paikallisesti samassa tietokoneessa.
Prototyyppi	tarkoittaa sovelluksen tai tietojärjestelmän versiota, jolla on tarkoitus varmistaa konseptin tekninen toimivuus ja/tai karottaa käyttäjien tarpeita.
Tietojärjestelmä	on ihmisistä, tietojenkäsittelylaitteista, tiedonsiirtolaitteista ja ohjelmistoista koostuva järjestelmä, jolla tehostetaan tai helpotetaan jotain tietojen käsittelyn toimenpiteitä tai tehdään se ylipäättään mahdolliseksi.
Tietokanta	on tietotekniikassa käytetty termi tietovarastolle. Se on koelma tietoja, joilla on yhteys toisiinsa.

Dokumentissa esiintyviä teknisiä termejä ovat seuraavat:

Alusta	koostuu käyttöjärjestelmästä ja muista tietojärjestelmän käyttämistä varusohjelmistoista.
Doxygen	on luokkadokumentaation generointiohjelma C#-lähdekooditiedostoille.

FreeMind	on ajatuskarttasovellus.
GanttProject	on ajan- ja resurssienhallintaohjelma projekteille.
Git	on hajautettu versiohallintajärjestelmä.
LaTeX	on ladontaohjelmisto.
Lähdekoodi	on tekstimuotoista ohjelmointikielistä listausta.
Trac	on hyvin yksinkertainen WWW-pohjainen alun perin ohjelmistoprojektien tehtävien hallintaan tarkoitettu tietojärjestelmä.
WP8	on lyhenne sanasta Windows Phone 8.
YouSource	on Git-versiohallintaohjelmistoa tukeva lähdekoodien julkistusjärjestelmä.

3 Taustaa ja tavoitteiden toteutuminen

Luvussa käsitellään projektin taustaa, projektissa toteutetulle tietojärjestelmän prototyypille ja muille tuloksille asetettujen tavoitteiden toteutumista sekä ryhmän jäsenille asetettujen oppimistavoitteiden toteutumista. Projektin tilaajan prototyypille asettamat välttämättömät ja tärkeät vaatimukset saatiin toteutettua. Projektin vaatimusmäärittelyssä [9] projektissa toteutettaviksi vaatimuksiksi rajatuista 142 vaatimuksesta kaksi jäi keskeneräiseksi ja yhtä ei toteutettu. Nämä kolme vaatimusta sovittiin jatkokehitykseen. Toteutetulla prototyypillä pystyttiin osoittamaan teknologiset mahdollisuudet uudelleenlaiselle hätäilmoitusjärjestelmälle, ja sillä voidaan havainnollistaa järjestelmän toimintaa. Projektissa kohdatut haasteet, niiden ratkaisut sekä projektiryhmän laatimat sovellusraportti ja vaatimusmäärittely antavat hyvät valmiudet järjestelmän jatkokehitykseen.

3.1 Taustaa ja tarpeita

Matkapuhelinteknologia on kehittynyt huimasti kuluneen vuosikymmenen aikana. Nykyisin Suomessa jokaiseen puhelinliittymään kytketään automaattisesti mahdollisuus tietoliikennedatan siirtämiseen, ja käytännössä suurin osa tänä päivänä myytävistä matkapuhelimista on varustettu "älyominaisuuksilla". Älypuhelimien laaja kattavuus tarjoaa luonnollisen ja tehokkaan tavan kehittää perinteistä hätäpuhelua nykyaikaan.

Hätäkeskuslaitos on ensimmäinen viranomaislenkki auttamisen ketjussa. Hätäkeskuksen tehtävänä on puhelun aikana luoda vakioidulla menettelyllä riskiarvio tilanteesta ja välittää tehtävä asianomaiselle viranomaiselle. Tänä päivänä hätäpuhelu on rajoittunut puheyhteyteen hätäkeskuksen päivystäjän kanssa. Pelkän puheyhteyden avulla kokonaisarvion tekeminen tilanteesta voi olla joskus erittäin haastavaa, joten virhearvioita ja niiden seurauksia voidaan lukea päivälehdistä aika-ajoin.

Virhearviot voivat aiheuttaa seuraavanlaisia ongelmia:

- Tilannetta tai riskiä ei tunnisteta, jolloin viranomaisapu viivästyy tai jää saapumatta, ja pahimmillaan menehtyy ihmishenkiä.
- Tilanne tai riski yliarvioidaan, josta seuraa viranomaisresurssien epätarkoituksen mukainen käyttö johtaen mm. seuraaviin ongelmiin:
 - Vaste on liian suuri tehtävän todelliseen riskiin nähden.

- Hälytysajojen lisääntyminen lisää liikenne-riskiä.

Hälyri-projektina aloitetun hankkeen tavoitteena on osoittaa uudenmallisten datayhteyksien laaja ja innovatiivinen hyödynnettävyys hätätilanteiden arvioinnissa ja hallinnassa. Hätäpuhelun aikana älypuhelimesta voitaisiin lähettää hätäkeskukseen ääni- ja videokuva, paikkatietoja sekä EKG- ja muuta anturidataa tukemaan riskiarvion tekemistä. Hätäkeskus voisi lähettää toimintaohjeita esimerkiksi videoina tai kuvina hädässä olevalle. Hätäkeskus voisi myös siirtää yhteyden suoraan auttajille.

Jyväskylän yliopiston tietotekniikan laitoksella on Sapporo-projektin kautta laaja kokemus julkisen hätäviestijärjestelmän kehittämisestä ja pilotoinnista. Siitä saatujen kokemusten [3] perusteella vastaava hätäilmoitusjärjestelmän pilotointi voitaisiin luotettavasti suorittaa yliopiston hankkeena. Hankkeessa kehitettävä tietojärjestelmä voisi toimia hätätilanteessa lisäapuna perinteisen 112-hätäpuhelun rinnalla.

Hankkeessa kehitettävän tietojärjestelmän valmistuttua jokainen kansalainen voisi ilmaiseksi ladata siihen sisältyvän mobiilisovelluksen omalle puhelimelleen niin halutessaan. Toiminta perustuu siis keskeisesti vapaaehtoisuuteen, eikä sovellus ensimmäisessä vaiheessa tule vaikuttamaan perinteisen 112-hätäpuhelun toimintaan ja siihen liittyviin säädöksiin.

3.2 Prototyypille asetettujen tavoitteiden toteutuminen

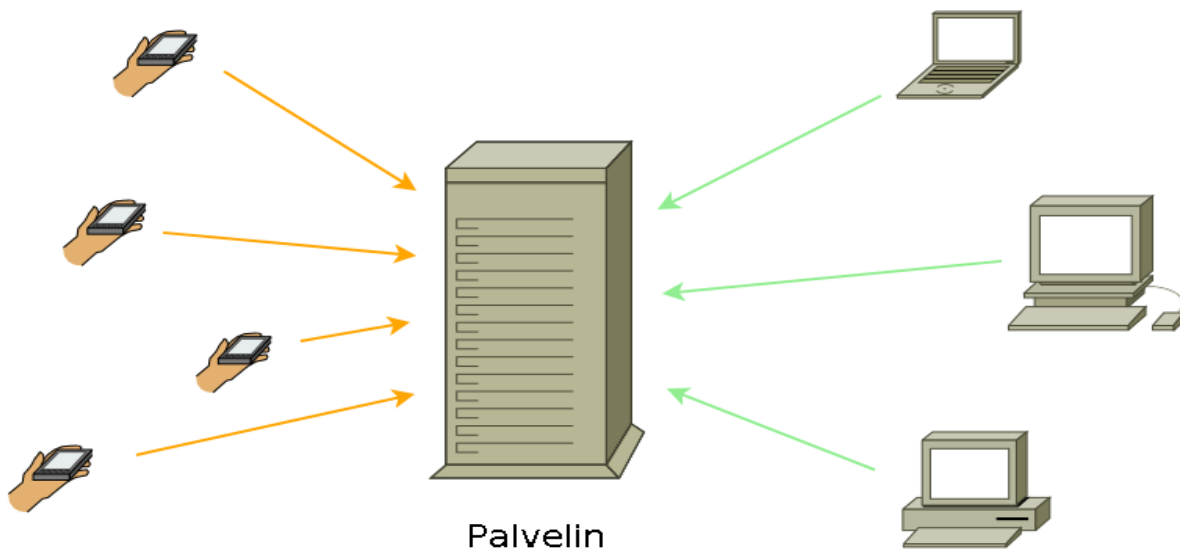
Tilaajan edustajat olivat määritelleet projektin alussa prototyypin kokonaistavoitteet, mutta tarkemmat vaatimukset selvitti projektiryhmä laatimalla kattavan vaatimusmäärittelyn. Vaatimusmäärittelyssä [9] vaatimukset priorisoitiin kuusiportaiselle asteikolle, jonka prioriteetit olivat välttämätön, tärkeä, projektissa tärkeä järjestelmässä mahdollinen, mahdollinen, idea ja ei toteuteta. Projektiryhmä kirjasi 105 vaatimusta hätäkeskusohjelmalle, 61 vaatimusta palvelimelle ja 115 vaatimusta älypuhelinsovellukselle. Projektissa toteutettaviksi vaatimuksiksi rajattiin vain tasojen välttämätön, tärkeä ja projektissa tärkeät vaatimukset. Toteutettavia vaatimuksia oli siten 25 hätäkeskusohjelmalle, 19 palvelinsovellukselle ja 98 älypuhelinsovellukselle.

Projektiryhmä täydensi vaatimusmäärittelyä uusilla vaatimuksilla, mutta projektissa toteutettaviksi rajattujen vaatimusten prioriteetteja ei jouduttu muuttamaan. Projektin aikana toteutettiin lähes kaikki projektissa toteutettavaksi rajatut vaatimukset. Vain kaksi vaatimusta jäi keskeneräiseksi ja yksi toteuttamatta. Projektissa sovellukseen kehitetyistä vaatimuksista toteutuneiksi on laskettu vaatimuskokonaisuudet, joiden kaikki välttämättömät vaatimukset on toteutettu.

Toteutetun tietojärjestelmän prototyyppi koostuu palvelinohjelmiston ohella yhdestä tai useammasta älypuhelinsovelluksesta ja hätäkeskussovelluksista (katso kuva 3.1) Älypuhelinsovelluksella voidaan kuvan 3.2 mukaisesti välittää hätäkeskussovellukseen palvelimen kautta ääntä ja videokuvaa, lähettää automaattisesti paikka- ja henkilötietoja, sekä välittää muuta tietoa, kuten esimerkiksi fysiologista mittausdataa (EKG). Älypuhelinsovellus huolehtii myös tarvittavasta kaistastaan huonoilla yhteyksillä.

Älypuhelinsovellus

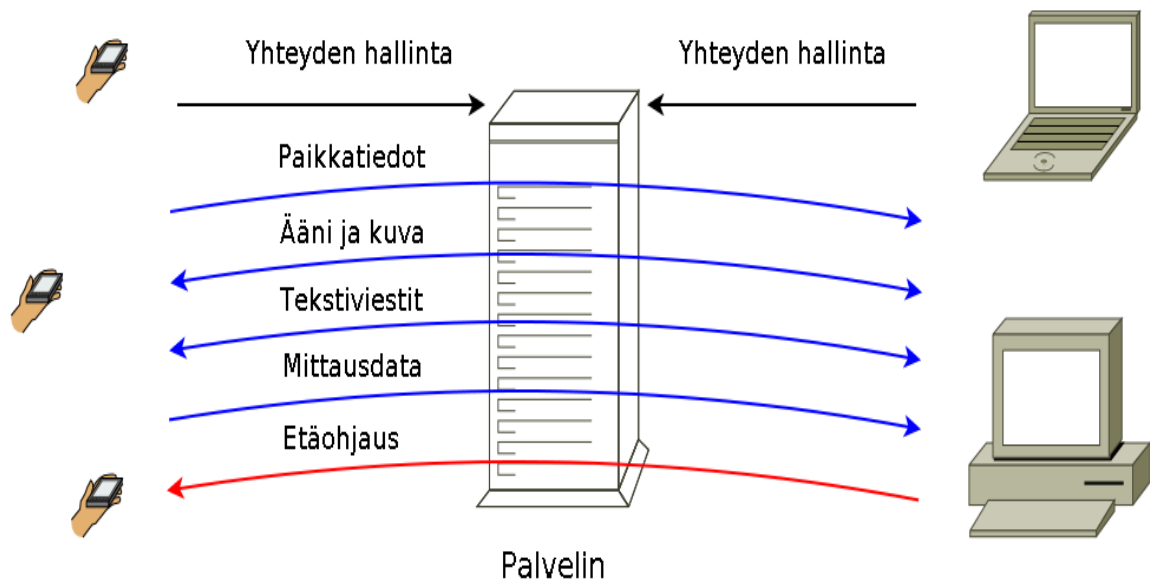
Hätäkeskussovellus



Kuva 3.1: Projektissa kehitetyn hätäilmoitusjärjestelmän prototyyppi.

Älypuhelinsovellus

Hätäkeskussovellus



Kuva 3.2: Prototyypin välittämät tiedot.

Hätäkeskuksen henkilökunnan käyttöön kehitetyllä työasemasovelluksella voidaan ottaa hätäilmoituksia käsiteltäväksi, jolloin sovelluksen käyttöliittymässä esitetään älypuhelimesta lähetetyt tiedot. Hätäilmoitussovelluksella voidaan pyytää älypuhelinsovelluksesta tietoja, kuten esimerkiksi fysiologista mittausdataa. Kehitetyllä prototyypillä voidaan siirtää vain generoitua fysiologista mittausdataa, koska projektiryhmä ei saanut käyttöönsä EKG-mittauslaitetta. Hätäkeskussovelluksella voidaan ohjata älypuhelinsovelluksen toimintoja, kuten esimerkiksi pyytää videokuvaa ja tarkentamaan sijaintia.

Hätäkeskussovellukseen toivottua ominaisuutta lähettää toimintaohjeita, kuvia ja videokuvaa älypuhelinsovellukselle ei toteutettu. Tietojärjestelmään toteutettu palvelinsovellus huolehtii hätäpuhelimien käsittelystä ja välittämisestä, muttei tavoitteisiin sisältyneestä tallentamisesta tietokantaan tai tiedostoon.

Toteutetulla prototyypillä pystytään osoittamaan teknologiset mahdollisuudet uudennlaiselle hätäilmoitusjärjestelmälle, ja sillä voidaan havainnollistaa järjestelmän toimintaa. Projektissa kohdatut haasteet ja niiden ratkaisut antavat hyvät lähtökohdat jatkokehitykselle. Projektissa kehitetty tietojärjestelmän prototyyppi on suunniteltu jatkokehitystä silmälläpitäen. Sovellusraportissa [12] on kuvattu tarkemmin prototyyppiä ja käytettyjä toteutusratkaisuja.

3.3 Oppimistavoitteiden toteutuminen

Ryhmän jäsenet olivat asettaneet seuraavia henkilökohtaisia oppimistavoitteita:

- Niko Mononen halusi oppia ohjelmistoprojektin johtamista.
- Veli-Mikko Puupponen halusi kokemusta projektimuotoisesta työtavasta ja ryhmätyöskentelystä.
- Ilkka Rautiainen halusi kehittää ohjelmointiosaamistaan.
- Atte Söderlund halusi oppia projektimuotoista työtapaa kokonaisuudessa.

Henkilökohtaiset oppimistavoitteet toteutuivat jokaisen ryhmän jäsenen osalta. Ryhmän jäsenet ymmärsivät hyvin oman roolinsa ja vastuunsa merkityksen sekä ajankäytön hallinnan merkityksen. Projektin varsinaisen läpiviennin lisäksi opiskelijat oppivat yhteistyötä ryhmän sisällä. Työskenteleminen ryhmässä vaati vastuunottamista ja -antamista. Pienryhmässä toimiessa myös erilaisiin ongelma- ja ristiriitatilanteisiin oli varauduttava. Näitä ongelmatilanteita ratkoessaan ryhmän jäsenet oppivat myös työelämässä tarvittavia kykyjä. Kukin ryhmän jäsen on analysoinut oppimaansa tarkemmin luvussa 9.

3.4 Tulokset ja niiden toteutuminen

Tietojärjestelmän ohella projektiryhmä toteutti seuraavat dokumentit:

- **Ajankäyttöraportti** sisältää ryhmän jäsenten kirjaamat työtunnit sekä niiden jakautumisen eri tehtäväkokonaisuuksille ja tehtäville.
- **Asennusohjeet** sisältävät tietojärjestelmän eri sovellusten asennusohjeet.
- **Esittelymateriaali** sisältää väli- ja loppuesittelyn materiaalit ja muistiot.
- **Esittelyviesti** pelastusalan asiantuntijoille sisältää lyhyen kuvauksen projektista ja siinä kehitetystä prototyypistä.
- **Esittelyvideo** [7] havainnollistaa kehitetyn prototyypin toimintaa.
- **Itsearvioinnit** sisältävät ryhmän jäsenten arvioinnit omasta toiminnasta, onnistumisesta, kokemuksista ja oppimisesta.
- **Lisenssisitoumuksella** ryhmän jäsenet vahvistavat sitoutuvansa sijoittamaan toteuttamansa lähdekoodin valitun avoimen lähdekoodin lisenssin alaisuuteen.

- **Luokkadokumentaatio [8]** sisältää lähdekoodista automaattisesti generoidut luokkia kuvaavat dokumentit.
- **Lähdekoodi** sisältää lähdekoodin kommentteineen.
- **Palaverien dokumentit** sisältävät palaverien esityslistat, pöytäkirjat ja tilakatsaukset.
- **Projektiraportti** kuvaa projektin toteutunutta läpivientiä ja asetettujen tavoitteiden saavuttamista sekä jäsenten kokemuksia ja oppimaa.
- **Projektisuunnitelma [10]** kuvaa projektin suunniteltuja tavoitteita, resursseja, käytänteitä, tehtävien työmääriä ja työnjakoa, aikataulua sekä riskien hallintaa.
- **Sovellusraportti [12]** kuvaa toteutetun sovelluksen käyttöliittymän, kokonaISRakenteen ja toiminnot, puutteelliset ja heikot toteutusratkaisut sekä jatkokehitysideat.
- **Sähköpostiarkistot** sisältävät kaikki projektin sähköpostilistoilla käydyt keskustelut.
- **Vaatimusmäärittely [9]** kuvaa projektin toiminnalliset ja tekniset vaatimukset.

Projektiryhmä laati kaikki suunnitellut dokumentit. Laadituista tuloksista esittelyviesti ja esittelyvideo eivät sisältyneet alunperin suunniteltuihin tuloksiin.

4 Organisaatio ja resurssit

Luvussa esitellään projektiorganisaatio, käytössä olleet resurssit sekä projektiin liittyvät oheiskurssit ja perehdytykset. Projektisuunnitelmaan kirjattu projektiorganisaatio ja resurssit toteutuivat, eikä merkittäviä muutoksia tullut.

4.1 Projektiorganisaatio

Projektiyhtymän muodostivat Niko Mononen, Veli-Mikko Puupponen, Ilkka Rautiainen ja Atte Söderlund. Niko Monosella oli aiempaa ohjelmointikokemusta työelämästä ja omien ohjelmistoprojektien kautta. Veli-Mikko Puupposella oli myös ohjelmointikokemusta yliopiston ohjelmointikurssien ja omien ohjelmistoprojektien kautta. Hänellä oli myös paljon tietämystä erilaisista tietojärjestelmistä, mistä oli hyötyä sovelluksen suunnittelu- ja toteutusvaiheessa. Ilkka Rautiaisella oli ohjelmointitaitoja yliopiston ohjelmointikursseilta. Atte Söderlundilla oli aiempaa kokemusta mobiiliohjelmoinnista, mikä osoittautui hyödylliseksi projektissa.

Tilaajan edustajina toimivat Tero Tuovinen ja Vesa Lappalainen tietotekniikan laitokselta. Jukka-Pekka Santanen toimi projektin vastaava ohjaaja. Teknisiä ohjaajia olivat Olli Kauppinen ja Jaakko Kosonen. Kosonen toimi ensisijaisena ohjaajana tietojärjestelmän toteutusratkaisuihin liittyvissä asioissa. Kauppisella oli tietämystä kohdealueesta ja älypuhelinsovellusten kehittämiseen liittyvää aiempaa kokemusta. Projektin sidosryhmiin kuului myös hätäkeskusten, pelastusalan ja ensihoidon asiantuntijoita, joista projektiin osallistuivat Helena Jäntti Kuopion yliopistollisesta sairaalasta ja Tapio Kettunen Kuopion hätäkeskuksesta.

Jyväskylän yliopiston ATK-lähituki vastasi ryhmän käytössä olevista laitteista ja ohjelmistoista. Projektiin kuuluivat viestintäkurssin kirjoitusviestinnän opettajista Timo Nurmi ja puheviestinnän opettajista Hanna Kivimäki. Johanna Silvennoinen piti projektiyhtymälle käytettävyysspäivän ja antoi palautetta prototyypin käyttöliittymistä.

Projektisuunnitelmaan kirjattu projektiorganisaatio toteutui. Helena Jäntin osallistuminen projektin läpivientiin oli tosin vähäistä. Tekniset ohjaajat ja tilaajan edustajat olivat kuitenkin hyvin tavoitettavissa. Projektiyhtymä olisi kenties voinut tukeutua hieman enemmän teknisiin ohjaajiin projektin aikana.

4.2 Projektin tilat ja laitteet

Projektiryhmä sai käyttöönsä tietotekniikan laitokselta Agorasta huoneen C223.2, jossa oli asennettuina neljä Windows 8 -työasemaa. Projektiryhmällä oli käytössä myös palvelin ja Lappalaisen projektiryhmälle lainaama Windows Phone 8 -puhelin.

Sovellusprojektien avotilassa oli ryhmän käytössä monitoimitulostin. Ryhmän jäsenet pystyvät tulostamaan projektiin liittyvät dokumentit ilman maksuja. Ryhmällä oli mahdollisuus varata käyttöönsä videoprojektori, kannettava PC ja digitaalisanelin. Projektin aikana näistä tarvittiin vain kannettavaa tietokonetta. Projektiryhmän jäsenillä oli käytössä myös omia Windows Phone -puhelimia ja Veli-Mikko Puupposen digitaalisanelin. Projektiryhmällä oli käytettävissä virkistystila, jossa oli vedenkeitin ja kahvinkeitin. Tietotekniikan laitos tarjosi ryhmälle kahvit, teet ja mehut.

Projektilla oli käytössään yhteinen verkkolevy ja WWW-sivusto projektin tiedostojen säilytystä varten. Verkkolevy oli hakemistossa `\\sovpa7.cc.jyu.fi\halyri` ja sivusto osoitteessa `http://sovellusprojektit.it.jyu.fi/halyri`.

Projektin julkistetut dokumentit sijoitettiin trac-wikiin URL-osoitteeseen `https://trac.cc.jyu.fi/projects/sovproj/wiki/Halyri`. Trac on hyvin yksinkertainen WWW-pohjainen alun perin ohjelmistoprojektien tehtävien hallintaan tarkoitettu järjestelmä. Lähdekoodi sijoitettiin Git-pohjaiseen YouSource-julkistusjärjestelmään URL-osoitteeseen `https://yousource.it.jyu.fi/halyri/`. Julkistetut dokumentit sijoitettiin pääosin suunnitellusti, vaikka joitakin dokumentteja sijoitettiin myös Google Drive -palveluun.

Projektilla oli käytössä sähköpostilistat `halyri@korppi.jyu.fi` ja `halyri_opetus@korppi.jyu.fi`. Sähköpostilistojen arkistoista `https://korppi.jyu.fi/kotka/servlet/list-archive/halyri/` ja `https://korppi.jyu.fi/kotka/servlet/list-archive/halyri_opetus/` keskustelua voi lukea WWW-selaimella kirjautumalla Korppi-opintotietojärjestelmään.

Pelastusalan asiantuntijoille perustettiin kolmaskin sähköpostilista nimeltään `app112.group@korppi.jyu.fi`, mutta sille lähetettiin ainoastaan kaksi viestiä.

4.3 Dokumentointityökalut

Projektisuunnitelma ja projektiraportti laadittiin $\text{\LaTeX}$ -ladontaohjelmistolla. Microsoft Office -toimisto-ohjelmistoilla laadittiin palaverien esitysgrafiikat ja lisenssisitoumus. Vaatimusmäärittelyn tekemisessä käytettiin FreeMind-ajatuskarttaohjelmaa. Palaverien pöytäkirjat laadittiin joko Microsoft Word -ohjelmistolla tai $\text{\LaTeX}$ -ladontaohjelmistolla. Projektin aikataulu laadittiin GanttProject-ohjelmalla ja tehtävien työtunnit Excel-taulukkolaskentaohjelmalla. Raakatekstimuodossa laadittiin muut tekstidokumentit, kuten esityslistat. Luokkadokumentit generoitiin lähdekoodista Doxygen-työkalulla.

Ajankäytön kirjaamiseen ja seurantaan ryhmällä oli käytössään Petri Heinosen sovellusprojekteille toteuttama Excel-sovellus [1]. Lisäksi kyseisestä sovelluksesta saatiin tilakatsauksissa vaadittavat esitysgrafiikat.

Dokumentointityökalujen käyttö toteutui suunnitellusti. Ajankäyttöseurantasovelluksen kanssa oli aluksi hieman hankaluuksia, ja se vaikeutti hieman projektin seurantaa, mutta ei vaikuttanut projektin tuloksiin.

4.4 Ohjelmointityökalut

Sovellukset toteutettiin Microsoftin Visual Studio 2013 -ohjelmalla ja Windows Phone SDK -kehitystyökaluilla [5]. Käytettyjä kirjastoja ovat Speex, SignalR sekä Microsoftin ASP.NET Framework ja Entity Framework. Ohjelmointikielenä käytettiin Microsoftin C#:a.

Suunniteltua tietokannanhallintajärjestelmää ei käytetty projektissa, koska prototyyppiin ei kehitetty tietokantaa. Käytetyt ohjelmointityökalut eivät olleet ryhmän jäsenille ennestään tuttuja, joten niihin tutustumiseen käytettävä aika oli huomioitu projektisuunnitelmassa. Työkaluja opittiin kuitenkin käyttämään hakemalla tietoa Internetistä ja teknisten ohjaajien avustuksella.

4.5 Luennot ja perehdytykset

Projektin rinnalla järjestettiin kaksi oheiskurssia: *Sovellusprojektin hallintaa, viestintää ja työkaluja* sekä *Projektiviestintä IT-alalla*. Oheiskurssien työtunnit kirjattiin omalle tehtäväkokonaisuudelle työajanseurantasovelluksessa.

Kurssiin *Sovellusprojektin hallintaa, viestintää ja työkaluja* sisältyivät seuraavat luennot:

- aloitusluento,
- vaatimusmäärittely,
- projektin johtaminen ja hallinta,
- käytettävyysspäivä,
- tekijänoikeus ja sopimukset sekä
- versiohallinta.

Kurssiin *Projektiviestintä IT-alalla* kuuluivat puhe- ja kirjoitusviestinnän luentoja ja ryhmätöiden ohella projektissa laadittujen dokumenttien kirjoitusasun ja rakenteen muokkauksen työtunnit. Projektin aikana järjestetyt kaksi väliesittelyä kuuluivat myös viestintäkurssiin.

Projektin aikana pidettiin kaikki projektisuunnitelmaan kirjatut luennot ja opetustapaamiset. Ryhmän jäsenet saivat järjestetyistä opetustapahtumista hyödyllistä tietoa projektityöskentelyyn ja etenkin käytettävyysspäivästä saatu palaute oli erittäin hyödyllistä. Ryhmän jäsenet olisivat kaivanneet hieman enemmän perehdytystä projektin hallinnasta ja suunnittelusta.

5 Projektin käytänteet

Luvussa kuvataan projektissa noudatettuja käytänteitä. Nämä käytänteet edesauttoivat projektin läpivientiä, asetettujen tavoitteiden saavuttamista ja tulosten toteuttamista. Käytänteillä varmistettiin, että projekti eteni aikataulussa sekä projektin aikana toteutetut tulokset ovat korkealaatuisia ja keskenään yhteneviä.

Käytänteet poikkesivat suunnitelluista tiedostojen nimeämisen, testauksen ja tulosten luovuttamisten osalta. Tiedostojen nimeämisen käytänteet eivät toteutuneet projektiryhmän sisäisissä versioissa. Suunniteltua lähdekoodiin kirjoitettua yksikkötestausta tehtiin hyvin vähän, sillä testattavia funktioita oli vain muutamia. Lisäksi järjestelmätestaussuunnitelma oli samalla testausraportin dokumenttipohja, ja tilaajan edustaja halusi tulokset USB-tikulla CD-levyn sijaan.

5.1 Tiedotus

Projektin tiedotuksesta projektiorganisaation sisällä vastasi ensisijaisesti projektipäällikkö. Ryhmän jäsenet kommunikoivat ensisijaisesti projektipäällikön kanssa, mutta välittivät kuitenkin vastuualueillaan oleelliset asiat eteenpäin koko projektiorganisaatiolle. Projektiorganisaatiolle ryhmä tiedotti vähintään projektissa kohdatuista haasteista ja vaihtoehtoista sekä tehdyistä valinnoista, ratkaisuista ja päätöksistä.

Projektiorganisaation sisäinen sähköpostilista `halyri@korppi.jyu.fi` oli tarkoitettu tilaajien edustajille ja ohjaajille suunnattuun tiedotukseen. Sähköpostilistalle kuuluvat kaikki projektiorganisaation jäsenet. Postilistalle lähetetyt viestit tallentuivat sähköpostiarkistoon, joka löytyy osoitteesta `https://korppi.jyu.fi/kotka/servlet/list-archive/halyri`.

Projektiryhmän jäsenten ja ohjaajien käytössä oli myös sähköpostilista `halyri_opetus@korppi.jyu.fi`. Sen sähköpostiarkisto sijaitsee osoitteessa `https://korppi.jyu.fi/kotka/servlet/list-archive/halyri_opetus/`. Sähköpostilistalla käsiteltiin sellaisia asioita, jotka eivät olleet merkityksellisiä tilaajalle. Tällaisia asioita olivat esimerkiksi projektiryhmän sisäiset palaverit sekä erilaiset opetukseen ja koulutukseen liittyvät asiat.

Ryhmän sisäinen tiedotus tapahtui pääosin suullisesti, sillä ryhmän jäsenet työsken-

telivät samassa tilassa ja olivat yhteydessä toisiinsa. Tarvittaessa tiedotus ja keskustelu voitiin ryhmän sisällä hoitaa sähköpostitse.

Toteutunut tiedotus vastasi suunnitelmaa. Projektiryhmä lähetti kysymyksiä ja tarkennuksia sähköpostilistoille. Projektiryhmä keskusteli projektihuoneessa projektin läpivientiin sekä sovelluksen kehitykseen ja suunnitteluun liittyvistä asioista. Projektiryhmän jäsenillä oli eriävät työajat, joten keskustelua käytiin myös paljon sähköpostilla ja pikaviestimillä.

Projektin vastaava ohjaaja olisi toivonut säännöllisempää ja aktiivisempaa tiedotusta palaverien välillä. Valituista toteutusratkaisuistakin olisi voinut tiedottaa enemmän palaverien ulkopuolella.

5.2 Palaverit

Projektiorganisaatio pyrki pitämään yhteisen palaverin kerran kahdessa viikossa. Seuraavan palaverin ajankohta päätettiin aina edellisessä palaverissa.

Palavereissa käsiteltiin edellisen palaverin jälkeen tapahtuneita projektin etenemiseen vaikuttaneita asioita sekä tulevia toimenpiteitä ja tarvittavia päätöksiä. Jokaisessa palaverissa käytiin läpi edellisen palaverin pöytäkirjaan merkityt päätökset sekä osallistujille osoitetut toimenpiteet ja niiden tila. Palavereissa projektipäällikkö esitti tilakatsauksen, jossa esitettiin projektiryhmän viikoittainen ajankäyttö eri tehtäväkonaisuuksittain.

Palavereissa käsiteltiin myös toteutettavan sovelluksen ominaisuuksia ja vaatimuksia, sekä niiden toteutusratkaisuja. Palavereissa käsitellyt asiat pyrittiin käymään läpi niin perusteellisesti, että asiakkaan edustajat ja projektiryhmän jäsenet ymmärsivät asiat samalla tavalla, eikä väärinymmärryksiä päässyt syntymään. Jos projektiryhmällä oli esittää sovelluksesta konkreettisia käyttöliittymään liittyviä demoja tai prototyyppejä, ne esiteltiin palavereissa. Palavereissa sovittiin myös projektiin liittyvistä käytänteistä.

Ryhmän jäsenet toimivat vuorollaan palavereissa sihteerinä tai puheenjohtajana siten, että kyseiset tehtävät kiersivät ryhmän keskenään sopimassa järjestyksessä. Puheenjohtaja johti keskustelua ja piti huolen siitä, että palaveri eteni esityslistan osoittamalla tavalla. Sihteeri laati palaverista pöytäkirjan, jonka hän toimitti myöhemmin palaverin puheenjohtajalle tarkastettavaksi. Kun pöytäkirja oli puheenjohtajan

osalta hyväksytty, puheenjohtaja tai sihteeri toimitti sen koko projektiorganisaatiolle. Tämän lisäksi jokaisen ryhmän jäsenen laatima ensimmäinen pöytäkirja toimitettiin myös vastaavan ohjaajan ja kirjoitusviestinnän opettajan esitarkastettavaksi. Pöytäkirja hyväksytettiin seuraavassa palaverissa, ja siihen voitiin tällöin esittää muutoksia.

Palaverien suunnitellut käytänteet toteutuivat projektissa. Muutamien pöytäkirjojen laatiminen ja tarkastaminen viivästyi useita päiviä. Tämä johtui ryhmän jäsenen kokemattomuudesta laatia pöytäkirjoja ja muiden töiden kasautumisesta.

5.3 Tiedostojen nimeäminen

Lähdekooditiedostojen nimeämisessä käytettiin C#-kielen yleisiä käytänteitä. Tiedostojen ja hakemistojen nimet kirjoitettiin englanniksi. Lisäksi tiedostonimet kirjoitettiin aina pienillä kirjaimilla ja välilyönnit korvattiin alaviivoilla (_). Lähdekoodissa ja dokumentoinnissa käytettyjen tekstitiedostojen tallennusmerkistönä käytettiin ensisijaisesti UTF-8-koodausta.

Dokumenttitiedostot nimettiin projektin nimen ohella niiden sisältöä kuvaavilla nimillä. Julkistettaessa nimiin lisättiin versionumero luvussa 5.6 esitettyjen käytänteiden mukaisesti. Esimerkiksi projektisuunnitelman pdf-tiedosto nimettiin `halyri_projektisuunnitelma_[numero].[numero].[numero].pdf`.

Projektin aikana julkistetut tiedostot nimettiin pääsääntöisesti suunnitellun mukaisesti. Ryhmän sisäisessä nimeämisessä ja numeroinnissa poikettiin kuitenkin suunnitellusta.

5.4 Hakemistorakenne

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

```
dokumentit
  ajankaytto
  esittelyt
  itsearviointit
```


- projektiraportti
- projektisuunnitelma
- sovellusraportti
- vaatimusmaarittely
- lissenssisitoumus
- palaverit
 - esityslistat
 - materiaalit
 - poytakirjat
 - tilakatsaukset
 - katselmoinnit
- testaus
 - testausraportit
 - testaussuunnitelmat
- sahkopostiarkistot
 - halyri
 - halyri_opetus
- software
 - class_documentation
 - installation_documents
 - installation_files
 - source_code
- esittelyviesti
- esittelyvideo

Hakemistorakenne toteutui suunnitellusti.

5.5 Lähdekoodi

Projektissa toteutettu lähdekoodi sijoitettiin YouSource-julkistusjärjestelmään. Sovelluksen lähdekoodi kirjoitettiin noudattaen yleisiä C# -koodin käytänteitä [6]. Lähdekoodin kommentoinnissa käytettiin C#-n XML-tyylistä käytäntöä. Kommentoinnissa käytettiin XML-tyyliä, jotta lähdekoodista voitiin generoida luokkadokumentaatio Doxygen-ohjelmalla. Lähdekoodissa käytetyt aliohjelmat, luokat ja muuttujat nimettiin mahdollisimman kuvaavilla englanninkielisillä nimillä. Myös koko lähdekoodin kommentointi toteutettiin englanniksi. Usean sanan nimet kirjoitettiin Ca-

melCasena ja lähdekoodiin sallittiin enintään 80 merkkiä riviä kohden.

Seuraavassa on esimerkki edellä esitettyjen käytänteiden mukaisesta C# -koodista.

```
/// Copyright (C) 2014 Veli-Mikko Puupponen, Ilkka Rautiainen
///
/// The MIT License (MIT)
///
/// Permission is hereby granted, free of charge, to any person obtaining a copy
/// of this software and associated documentation files (the "Software"), to
/// deal in the Software without restriction, including without limitation the
/// rights to use, copy, modify, merge, publish, distribute, sublicense,
/// and/or sell copies of the Software, and to permit persons to whom the
/// Software is furnished to do so, subject to the following conditions:
///
/// The above copyright notice and this permission notice shall be included in
/// all copies or substantial portions of the Software.
///
/// THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
/// IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
/// FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
/// AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
/// LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
/// FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS
/// IN THE SOFTWARE.

using System;
using HalyriServer.Model.Transfer;
using HalyriServer.Services.Exceptions;

namespace HalyriServer.Controllers
{
    /// <author>Veli-Mikko Puupponen, Ilkka Rautiainen</author>
    /// <summary>
    /// The singleton class manages active emergency connections (Connection).
    /// It contains a dictionary for active connections and their
    /// identifying GUIDs. All emergency connection states and life cycle
    /// methods are handled by the class. It functions as a broker between
    /// the mobile emergency connection client and the call center client.
    /// </summary>
    public class ConnectionController
    {
        private static readonly Lazy<ConnectionController> instance =
            new Lazy<ConnectionController>(() => new ConnectionController());
        private DatabaseController databaseController;
```

```
    /// <summary>
    /// Returns the ConnectionController instance.
    /// </summary>
    public static ConnectionController Instance
    {
        get { return instance.Value; }
    }

    /// <summary>
    /// The method generates a new GUID. It currently uses
    /// Guid.NewGuid() of .NET framework. No collision checking
    /// is performed. All-zero GUIDs are not returned.
    /// </summary>
    /// <returns>A new GUID formatted as a string.</returns>
    /// <example>
    /// <pre name="test">
    /// GetNewGuidStrig().length() > 0 === true;
    /// GetNewGuidStrig().length() < 25 === true;
    /// GetNewGuidStrig().length() == 25 === false;
    /// </pre>
    /// </example>
    public string GetNewGuidStrig()
    {
        Guid guid;
        do
        {
            guid = Guid.NewGuid();
            while (guid == Guid.Empty);
            return guid.ToString();
        }
    }
}
```

Lähdekoodin ja kommentoinnin käytänteet toteutuivat suunnitelman mukaisesti. Kommentoinneissa oli kuitenkin puutteita, jotka korjattiin lähdekoodien viimeisimpään versioon.

5.6 Versiointi ja versiohallinta

Sovelluksen lähdekoodi sijoitettiin Git-pohjaiseen YouSource-julkistusjärjestelmään, josta se on myös tilaajan edustajien ja ohjaajien saatavilla. Kaikki sovellukseen ja projektiin liittyvä julkistettu dokumentaatio säilytetään trac-wikissä.

Dokumenttien versioissa käytettiin kolmiportaista versionumerointia. Ryhmän sisäiset versiot aloitettiin versionumerosta 0.0.1, ja kunkin uuden version osalta kasvatettiin vähiten merkitsevää numeroa yhdellä. Tällöin toinen versio oli versionumeroltaan 0.0.2. Projektiorganisaatiolle julkistettujen versioiden numerointi aloitettiin versionumerosta 0.1.0. Seuraavat versiot numeroitiin kasvattamalla toisen tason numeroa yhdellä. Ensimmäisen hyväksytyn version numero oli 1.0.0, ja sitä seuraavissa hyväksytyissä versioissa kasvatettiin toisen tason numeroa yhdellä.

Suunnitellussa versiohallinnassa pitäydyttiin, ja julkistettujen dokumenttien versionumerointi vastasi suunniteltua.

5.7 Testaus

Toteutettujen sovellusten toiminnan verifiointi tapahtui yksikkö- ja järjestelmätestauksella. Yksikkötestausta tehtiin vain vähän, ja se suoritettiin testaamalla yksittäiset luokat ja metodit ComTestin C#-versiolla [4]. Yksikkötestit laati ja suoritti aliohjelman tekijä.

Järjestelmätestaukseen osallistuivat suunnitelmasta poiketen ainoastaan Puupponen ja Rautiainen. Testauksen vastuuhenkilö laati ryhmän kanssa järjestelmän testaussuunnitelman vaatimusmäärittelyn pohjalta. Testaussuunnitelma oli samalla testausraportin dokumenttipohja. Kukin testausraportti siten sisältää testauskerroilla suoritettujen testitapausten ja testauskerralla suoritettujen testitapausten tulokset, virheet ja mahdolliset puutteet.

Järjestelmätestauskerroilla havaitut virheet ja puutteet raportoitiin testausraportteihin [13], [11], [14] ja [15] sekä korjattiin. Testauksissa havaituista virheistä ja puutteista sekä niiden ratkaisusta keskusteltiin tilaajan edustajien kanssa viimeisessä palaverissa. Testauskertoja oli yhteensä viisi ja neljällä ensimmäisellä testauskerralla havaittujen virheiden ja puutteiden korjausten jälkeen suoritettiin vielä viimeinen regressiotestaus. Regressiotestauksen tulokset kuvattiin testausraportissa [16].

Vastaava ohjaaja koekäytti tietojärjestelmän molempia käyttöliittymiä yhden kerran projektihuoneessa. Vastaava ohjaaja antoi projektiryhmälle palautetta [17] huomautuksista puutteista ja kehitysideoista. Suunniteltua useamman henkilön koekäyttöä ei suoritettu.

Suunnitellusta poiketen testaussuunnitelma oli samalla testausraporttipohja, ja lähdekoodiin kirjoitettua yksikkötestausta tehtiin hyvin vähän. Muilta osin sovellusten testaus toteutui suunnitellusti.

5.8 Tulosten katselmoinnit ja hyväksyminen

Ryhmän projektissa kirjoittama lähdekoodi katselmoitiin kaksi kertaa projektin aikana. Ensimmäiseen katselmointiin osallistui teknisten ohjaajien lisäksi koko projektiryhmä, vastaava ohjaaja ja tilaajan edustajista Vesa Lappalainen. Toiseen katselmointiin osallistui tilaajan edustajista Lappalainen, teknisistä ohjaajista Kosonen ja vastaava ohjaaja, sekä projektiryhmän jäsenistä Puupponen, Rautiainen ja Söderlund. Katselmoinneissa tekninen ohjaaja kommentoi lähdekoodia antaen vinkkejä ja parannusehdotuksia. Tekninen ohjaaja myös hyväksyi lähdekoodin viimeisessä katselmoinnissa. Projektiryhmä kirjasi katselmointien havainnot muistioiksi.

Projektin lopussa tulokset kokonaisuutena hyväksyttiin projektin ohjaajilla ja tilaajan edustajilla. Yksittäisistä tuloksista tilaajan edustajan hyväksyntä tarvittiin vähintään toteutetuille sovelluksille, sovellusraportille ja vaatimusmäärittelylle. Tekninen ohjaaja hyväksyi lähdekoodin yhdessä tilaajan edustajan kanssa. Vastaava ohjaaja hyväksyi projektin keskeisimmät dokumentit, joita ovat projektisuunnitelma, projektiraportti, sovellusraportti, vaatimusmäärittely ja luokkadokumentit. Projektisuunnitelma ja projektiraportti, sovellusraportti sekä vaatimusmäärittely hyväksyttiin projektipäällikön, tilaajan edustajan ja projektin vastaavan ohjaajan allekirjoituksilla.

Katselmoinnit ja tulosten hyväksyminen toteutuivat pääosin suunnitelman mukaisesti. Mononen ei päässyt osallistumaan toiseen lähdekoodin katselmointiin. Katselmoinnit auttoivat ryhmän jäseniä kehittämään ohjelmointitaitojaan saatuaan rakentavaa palautetta kirjoittamistaan koodista.

5.9 Tulosten koostaminen ja toimittaminen

Projektiryhmä kokosi projektin tulokset projektikansioon ja CD-levylle. Projektikansioon kerättiin kaikki projektissa laaditut dokumentit ja lähdekoodi. Lisäksi sähköpostiarkistot, tiivistetty projektin kuvaus ja jäsenten itsearvioinnit liitettiin projektikansioon ja CD-levylle. CD-levylle tallennettiin edellisten lisäksi myös kehitetty ajettava sovellus ml. asennuspaketti ja asennusohjeet.

Tulokset toimitettiin tilaajalle USB-tikulla. Laitokselle toimitettiin projektikansio, joka sisältää projekti-CD:n. Toinen CD-levy toimitettiin laitoksen arkistoon. CD-levyt ja USB-tikut koostettiin vasta, kun kaikki projektin tulokset oli hyväksytty. Projektikansio sijoitettiin projektitilan kokoushuoneessa olevaan kirjahyllyyn.

Projektissa käytetty palvelin siirrettiin projektin päätyttyä tilaajan haltuun. Tietojärjestelmän viimeisimmät versiot ovat saatavilla YouSourcesta osoitteesta

<https://yousource.it.jyu.fi/halyri/>.

Tulosten koostaminen ja toimittaminen toteutuivat suunnitelman mukaisesti paitsi, että tilaajan edustajat halusivat tulokset USB-tikulla CD-levyn sijaan.

6 Tehtävät, työmäärät ja työnjako

Luvussa esitetään projektiryhmän jäsenten suunnitellut ja toteutuneet vastuualueet, tehtävät ja työnjako. Lisäksi luvussa esitetään ryhmän jäsenten arvioidut ja toteutuneet työtunnit tehtäväkokonaisuuksittain.

Toteutuneet työmäärät ylittivät noin 500 tunnilla suunnitellut. Puupponen käytti noin 240 tuntia ja kukin muista jäsenistä vähintään 60 tuntia suunniteltua enemmän työtunteja. Kokonaisuutena toteutunut työmäärä tehtäväkokonaisuuksittain vastaa prosenttiosuuksiltaan suunniteltua, vaikka käytettyjä tunteja olikin huomattavasti suunniteltua enemmän.

Jäsenten välinen työtehtävien jako ei vastannut suunnitelmaa. Vähän tai ei lainkaan ohjelmoineet henkilöt keskittyivät projektin hallintaan sekä ongelmien tutkimiseen ja järjestelmätestaukseen.

6.1 Vastuualueet

Projektipäällikkönä toimi Niko Mononen ja varapäällikkönä Atte Söderlund. Varapäällikkö vastasi projektipäällikön tehtävistä silloin, kun projektipäällikkö ei jostain syystä pystynyt hoitamaan jotain edellä mainituista tehtävistä. Varapäällikkö joutui muutaman kerran vastaamaan projektipäällikön tehtävistä hänen menojen ja kiireiden vuoksi.

Projektipäällikön vastuulle kuuluivat projektin suunnittelu ja hallinta sekä ajankäytön seuranta. Projektipäällikkö vastasi myös projektisuunnitelman ja projektiraportin laatimisesta. Projektipäällikön tehtäviin kuului lisäksi projektin tilan seuranta ja poikkeamiin reagoiminen.

Ryhmä jakoi tehtäviä monipuolisesti siten, että kaikki jäsenet tekivät ainakin jossain vaiheessa projektia suunnittelua, toteutusta ja testausta. Tällöin kukin jäsen sai kokemusta em. ohjelmistokehitysprosessin tehtäväkokonaisuuksista. Tuloksille nimetyt vastuuhenkilöt on esitetty taulukossa 6.1.

Vastuuhenkilö vastasi kyseisen tuloksen etenemisestä ja laadusta. Dokumentointia toteutettiin koko projektin ajan tasaisesti, jotta kaikki oleellinen saatiin kirjattua. Dokumentit laadittiin koko projektiryhmän voimin siten, että kukin laati vähintään yhden dokumentin.

Tulos	Suunniteltu	Toteutunut
Hätäkeskussovellus	Ilkka Rautiainen	Atte Söderlund
Käyttöliittymät	Atte Söderlund	Atte Söderlund
Mobiilisovellus	Atte Söderlund	Atte Söderlund
Palvelinsovellus	Veli-Mikko Puupponen	Veli-Mikko Puupponen
Projektisuunnitelma	Niko Mononen	Niko Mononen
Projektiraportti	Niko Mononen	Niko Mononen
Sovellusraportti	Atte Söderlund	Ilkka Rautiainen
Testausdokumentit	Ilkka Rautiainen	Ilkka Rautiainen
Tietokanta	Veli-Mikko Puupponen	–
Vaatimusmäärittely	Veli-Mikko Puupponen	Veli-Mikko Puupponen

Taulukko 6.1: Olennaisten tulosten suunnitellut ja toteutuneet vastuuhenkilöt.

Suunniteltu vastuuhenkilö vastasi vastuualueistaan koko projektin ajan. Suunniteltu vastuuhenkilöiden jako ei toteutunut. Tietokantaa ei prototyyppiin suunniteltu, eikä toteutettu. Hätäkeskussovelluksen ja sovellusraportin vastuuhenkilöiden vaihtamisen syynä oli Rautiaisen keskittyminen dokumentointiin sekä hätäkeskussovelluksen ja älypuhelinsovelluksen riippuvuus toisistaan. Vastuuhenkilöiden poikkeaminen suunnitellusta ei vaikuttanut projektin tuloksiin, eikä läpivientiin.

6.2 Työnositus ja työmäärä

Projektiryhmän suunniteltiin käyttävän Sovellusprojektin osalta projektiin 250 tuntia kutakin jäsentä kohden, jolloin koko ryhmän työtunnit olisivat olleet yhteensä 1000 tuntia. Kyseinen työmäärä olisi vastannut kymmentä opintopistettä ja olisi ollut noin 14 tuntia viikossa jäsentä kohden. Lisäksi kunkin projektiryhmän jäsenen suunniteltiin käyttävän 70 tuntia oheiskursseihin. Projektin työmäärän suunniteltiin jakautuvan tehtävittäin ja jäsenten kesken kuvien 6.1 ja 6.2 mukaisesti.

Tehtäväkokonaisuus	Tehtävä	NM		VMP		IR		AS		yht		tot
		s	t	s	t	s	t	s	t	s	t	
Projektin hallinta	Projektsuunnitelma	40	79	0	0	0	0	0	0	40	79	
	Seuranta ja hallinta	19	15	2	3	2	4	2	0	25	22	
	Suunnittelu ja raportointi	20	3	2	2	2	6	2	0	26	11	
	Projektiraportti	40	46	0	0	0	0	0	0	40	46	
	Lisenssisitoumus	1	1	1	1	1	1	1	1	4	4	
	Tiedotus	1	1	1	1	1	1	1	1	4	4	
	Viimeistely ja kokoaminen	20	30	0	0	0	0	0	0	20	30	
	Yhteensä	141	175	6	7	6	12	6	2	159	196	
Palaverit	Esityslistat	1	1	1	1	1	1	1	1	4	4	
	Palaverit	20	16	20	20	20	21	20	17	80	74	
	Pöytäkirjat	10	19	10	25	10	61	10	9	40	114	
	Valmistelu ja tutustuminen	8	16	8	6	8	6	8	10	32	38	
	Katselmoinnit	4	2	4	6	4	2	4	4	16	14	
	Yhteensä	43	54	43	58	43	91	43	41	172	244	
Perehtyminen	Aihealueeseen perehtyminen	5	0	5	20	5	17	5	3	20	40	
	Toteutukseen perehtyminen	20	19	20	38	20	27	20	5	80	89	
	Yhteensä	25	19	25	58	25	44	25	8	100	129	
Määrittely												
	Alustava vaatimusmäärittely	2	2	20	31	2	9	2	2	26	44	
	Vaatimusmäärittelyn päivitys	1	0	10	15	1	0	1	0	13	15	
	Vaatimusmäärittelyn viimeistely	0	0	2	5	0	0	0	0	2	5	
	Yhteensä	3	2	32	51	3	9	3	2	41	64	
Suunnittelu												
	Tietojärjestelmän rakenne	2	1	10	7	2	0	0	0	14	8	
	Tietokannat	1	0	10	4	0	0	0	0	11	4	
	Asetusten hallinta	1	0	10	2	1	1	1	1	13	4	
	Mobiilikäyttöliittymä	1	1	2	1	2	2	20	2	25	6	
	Hätäkeskuskäyttöliittymä	1	21	2	1	2	2	20	2	25	26	
	Palvelinsovellus	2	0	20	20	10	6	0	0	32	26	
	Rajapinnat	1	0	10	6	10	1	0	0	21	7	
	Yhteensä	9	23	64	41	27	12	41	5	141	81	

Kuva 6.1: Suunnitellut ja toteutuneet tunnit tehtävittäin 1/2.

Tehtäväkokonaisuus	Tehtävä	NM		VMP		IR		AS		yht	tot
		s	t	s	t	s	t	s	t	s	t
Toteutus	Tietojärjestelmän rakenne	0	0	10	10	5	14	0	3	15	27
	Tietokannat	1	0	10	10	0	0	0	0	11	10
	Asetusten hallinta	1	0	10	20	1	1	1	7	13	28
	Mobiilikäyttöliittymä	4	0	2	2	15	0	40	80	61	82
	Hätäkeskuskäyttöliittymä	4	16	2	2	15	1	40	80	61	99
	Palvelinsovellus	4	0	20	85	20	4	0	0	44	89
	Rajapinnat	4	0	10	124	20	35	5	97	39	256
	Yhteensä	18	16	64	253	76	55	86	267	244	591
Testaus	Suunnittelu	1	0	2	4	20	8	5	0	28	12
	Yksikkötestaus	2	1	5	1	5	1	5	1	17	4
	Järjestelmätestaus	2	0	2	2	10	26	5	0	19	28
	Yhteensä	5	1	9	7	35	35	15	1	64	44
Viimeistely	Sovellusraportti	3	0	2	26	20	69	11	0	36	95
	Lähdekoodin viimeistely	3	28	5	20	15	10	20	5	43	63
	Yhteensä	6	28	7	46	35	79	31	5	79	158
Tehtäväkokonaisuus	Tehtävä	NM		VMP		IR		AS		yht	tot
		s	t	s	t	s	t	s	t	s	t
Sovellusprojekti yhteensä		250	318	250	521	250	337	250	331	1000	1507
Tehtäväkokonaisuus	Tehtävä	NM		VMP		IR		AS		yht	tot
		s	t	s	t	s	t	s	t	s	t
Oheiskurssit	Puheviestintä	25	27	25	28	25	23	25	24	100	102
	Kirjoitusviestintä	25	21	25	25	25	22	25	21	100	89
	Sovellusprojektin hallinta	20	16	20	16	20	15	20	15	80	62
	Yhteensä	70	64	70	69	70	60	70	60	280	253
Sovellusprojektit ja oheiskurssit yhteensä		320	382	320	590	320	397	320	391	1280	1760

Kuva 6.2: Suunnitellut ja toteutuneet tunnit tehtävittäin 2/2.

Suunniteltu työnositus perustui kehitettävän tietojärjestelmän kokonaisrakenteeseen. Tehtävien jaossa huomioitiin jokaisen projektin jäsenen vastuualueet. Työmäärien suunnittelussa oli huomioitu kehitettävän tietojärjestelmän rakenne, projektin jäsenten taidot ja aikataulut sekä projektin aikataulu. Työmäärien järkevyyttä oli varmistettu vertailemalla Paatti-projektin projektiraportin [2] suunniteltuja ja toteutuneita työmääriä.

Projekti-suunnitelman tekemiseen arvioidut tunnit ylittyivät 39 tunnilla johtuen liian optimisesta arviosta suunnitteluvaiheessa. **Palaveri**ihin käytetyt työtuntimäärät ylittyivät 72 tunnilla suunnitelluista työtunneista, koska pöytäkirjojen laatiminen osoittautui aluksi hankalaksi ja työlääksi.

Perehtyminen työkaluihin ja projektin aihealueeseen ylitti hieman arvioidut työtunnit. Työtunteja oli hankala arvioida, koska ei ollut selkeää käsitystä siitä, kuinka monta uutta työkalua projektin jäsenet joutuvat opettelemaan. Työtunnit ylittyivät 29 tunnilla, joka on lähes kolmannes suunnitelluista työtunneista.

Koska kehitettävä tietojärjestelmä oli laaja, sen kartoittamiseen ja **vaatimusmäärittelyn** laatimiseen kului työtunteja 23 tuntia suunniteltua enemmän .

Suunnittelun työtunnit alittuivat 62 tunnilla arvioiduista työtunneista. Tämä johtuu lähinnä ryhmän jäsenten tavasta kirjata tehdyt työtunnit, sillä suurin osa suunnittelusta on kirjattu toteutuksen työtunteihin.

Toteutuksen toteutuneet työtunnit ylittyivät 347 tuntia. Tämä johtuu siitä, että suunnitelmaa laatiessa ei osattu ottaa huomioon kehitettävän alustan haasteellisuutta tarpeeksi hyvin. Puupponen käytti suunniteltuun verrattuna lähes nelinkertaisen tuntimäärän toteutukseen. Myös Söderlundin toteutukseen käyttämät työtunnit ylittyivät yli kolminkertaisesti.

Testauksen toteutuneet työtunnit alittuivat 20 tunnilla suunnitellusta Puupponen suoritti yhden testauskerran, mutta muuten testaus keskitettiin Rautiaiselle, koska yhden testauskerran suorittaminen vei vain puoli tuntia. **Viimeistelyn** toteutuneet työtunnit ylittivät suunnitellun 79 tunnilla. Viimeistelyyn jouduttiin käyttämään tuplasti työtunteja suunniteltuun verrattuna, koska lähdekoodien viimeistely ja sovellusraportin laatiminen osoittautuivat suunniteltua työläämmäksi.

Kun tarkastellaan sovellusprojektiin kokonaisuudessaan käytettyjä työtunteja, saadaan noin 507 tuntia suunniteltua enemmän. Tämä ylitys johtuu enimmäkseen toteutuksessa kohdatuista haasteista käytettyjen työkalujen ja alustan kanssa, sekä ryhmän jäsenten kokemattomuudesta laatia kattavia dokumentteja.

Työtehtävät keskittyivät joissain tapauksissa tietyille ryhmän jäsenille. Työtunneista nähdään selvästi Puupposen ja Söderlundin tehneen paljon toteutusta, kun taas Mononen vastasi projektin hallinnasta. Rautiainen toimi paljon sihteerinä sekä laati huomattavan määrän pöytäkirjoja sovellusraportin ja testaussuunnitelman lisäksi.

Työnjaossa onnistuttiin melko hyvin huomioimaan jäsenten poissaolot sekä muiden kurssien vaatima työmäärä. Tämä näkyy muutamien tehtävien keskittämällä yksittäiselle jäsenelle ja tehtävien siirroissa jäsenten kesken. Mononen teki projektin lopussa lähdekoodin **viimeistelyä** Söderlundin sijasta, Puupponen **ohjelmoi palvelinta** Rautiaisen sijaan ja **perehtyi** käytettäviin työkaluihin sekä Söderlund hoiti Rautiaisen osan **käyttöliittymien ohjelmoinnista**. Tarvittavia työtunteja ei osattu arvioida riittävän tarkasti, joten toteutuksen ja projektinhallinnan toteutuneet työmäärät ylittyivät suunnitellusta. Pöytäkirjojen laatiminen siirrettiin Rautiaiselle ja Puupponen laati noin neljänneksen sovellusraportista.

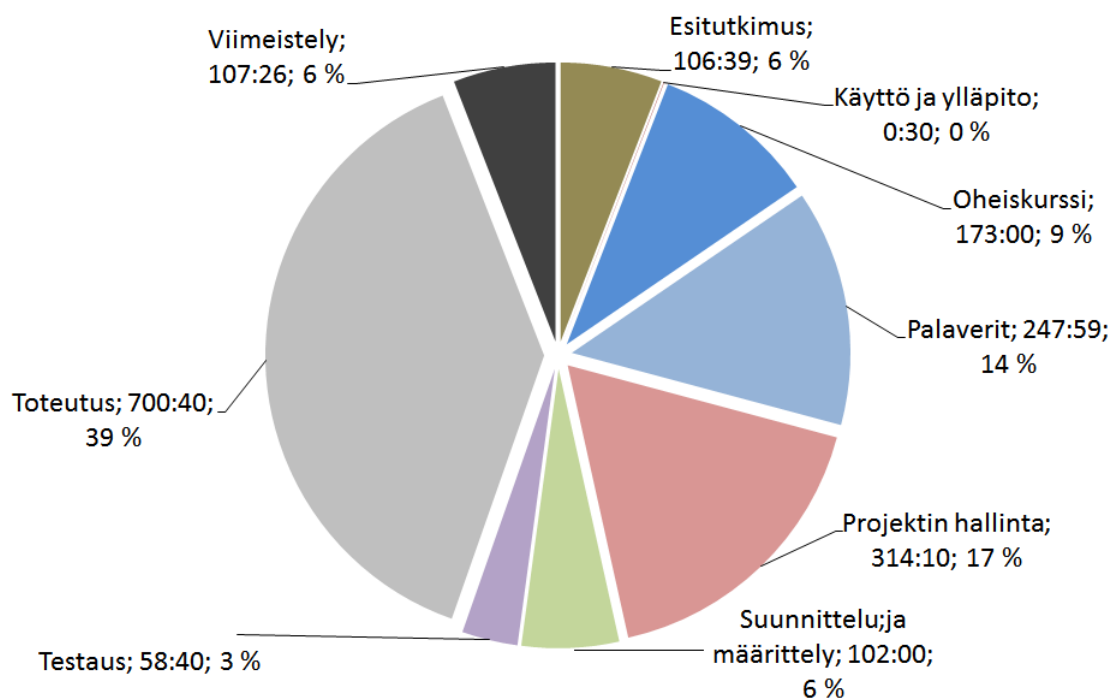
Toteutuneita työtunteja kertyi jäsenille vastuualueillansa huomattavasti suunniteltua enemmän, ja yksittäisten ryhmän jäsenten kokonaistyötuntimäärien erot olivat suuria. Puupponen käytti noin 200 tuntia enemmän kuin muut projektiryhmän jäsenet. Tämä johtui siitä, että toteutettavia sovelluksia oli monta, ja jokaisen sovelluksen rakenteen ymmärtäminen vaati huomattavan määrän työtunteja. Tämän lisäksi palvelinsovellus sekä videon- ja äänensiirto osoittautuivat hankaliksi suunnitella ja toteuttaa. Hankaluudesta johtuen työtä ei voinut helposti siirtää muille jäsenille, koska toteutuksen ymmärtäminen olisi vaatinut huomattavan määrän työtunteja. Projektin lopussa suoritettu lähdekoodien viimeistely ja dokumenttien laatiminen auttoivat hieman työmäärien tasaamisessa.

Oheiskursseihin käytettyjen työtuntien määrä on lähes sama kaikilla ryhmän jäsenillä. Pieniä yksilöllisiä eroja on kuitenkin esimerkiksi valmistautumiseen tarvittavissa tunneissa.

Ryhmän olisi kannattanut heti projektin alussa sopia tarkemmin työtuntien kirjaimiskäytännöt. Jos tunnit olisi kirjattu kaikissa kohdissa työtunneille suunniteltujen nimikkeiden alle, olisi suunniteltujen ja toteutuneiden tuntien vertailu ollut paljon helpompaa. Toteutuneissa työtunneissa on mukana myös suunnittelemattomia tehtäviä, kuten esimerkiksi esittelyvideon laatiminen.

6.3 Ryhmän työtunnit tehtäväkokonaisuuksittain

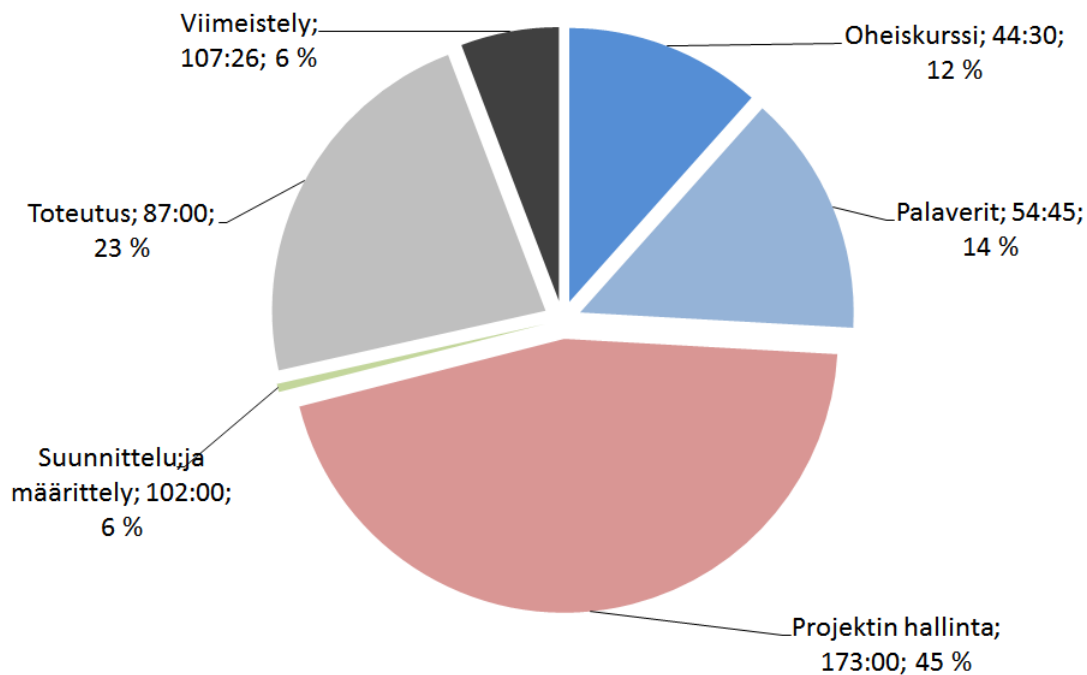
Piirakkagraafissa kuvassa 6.3 on esitelty työtuntien jakautuminen tehtäväkokonaisuuksittain. Toteutuksen suuri osuus työtunneista selittyy sillä, että siihen merkittiin myös tunteja, jotka olisivat kuuluneet pikemminkin suunnitteluun tai koodin viimeistelyyn. Sovelluksen kehityksen työtunneista oli hyvin vaikeaa erotella toteutuneita tunteja eri tehtäväkokonaisuuksiin ja usein koko päivän työtunnit merkittiinkin vain toteutukseen. Jos suunnitteluun käytettyjä tunteja ei olisi kirjattu toteutuksen alle, niin tehtäväkokonaisuuksien toteutuneiden työtuntien jako vastaisi hyvin suunniteltua jakoa, vaikkakin tehtyjen tuntien määrä ylitti suunnitellun työmäärään.



Kuva 6.3: Ryhmän työtunnit tehtäväkokonaisuuksittain.

6.4 Niko Monosen työtunnit tehtäväkokonaisuuksittain

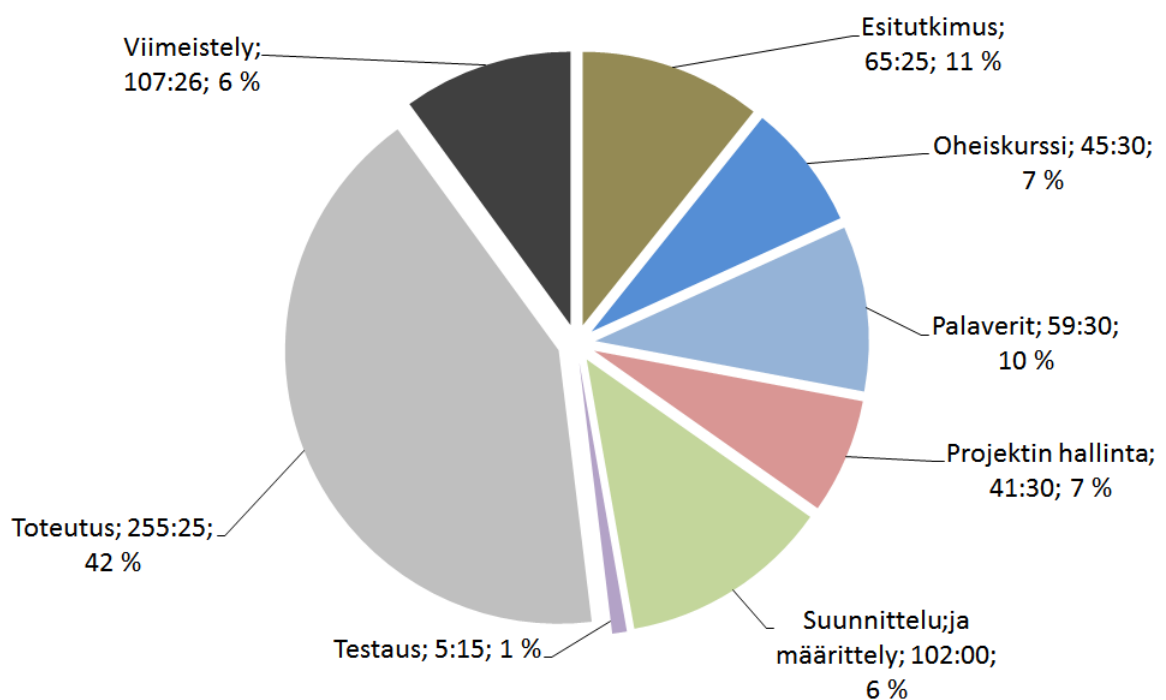
Mononen toimi projektipäällikkönä koko projektin ajan, mikä näkyy selvästi projektin hallinnan suurena osuutena kuvassa 6.4. Suurin osa työtunneista projektin aikana kului projektisuunnitelman ja projektiraportin kirjoittamiseen. Järjestelmän laajuuden vuoksi ja suunnitelmasta poiketen joutui projektipäällikkö myös tekemään käytännön toteutusta.



Kuva 6.4: Niko Monosen työtunnit tehtäväkokonaisuuksittain.

6.5 Veli-Mikko Puupposen työtunnit tehtäväkokonaisuuksittain

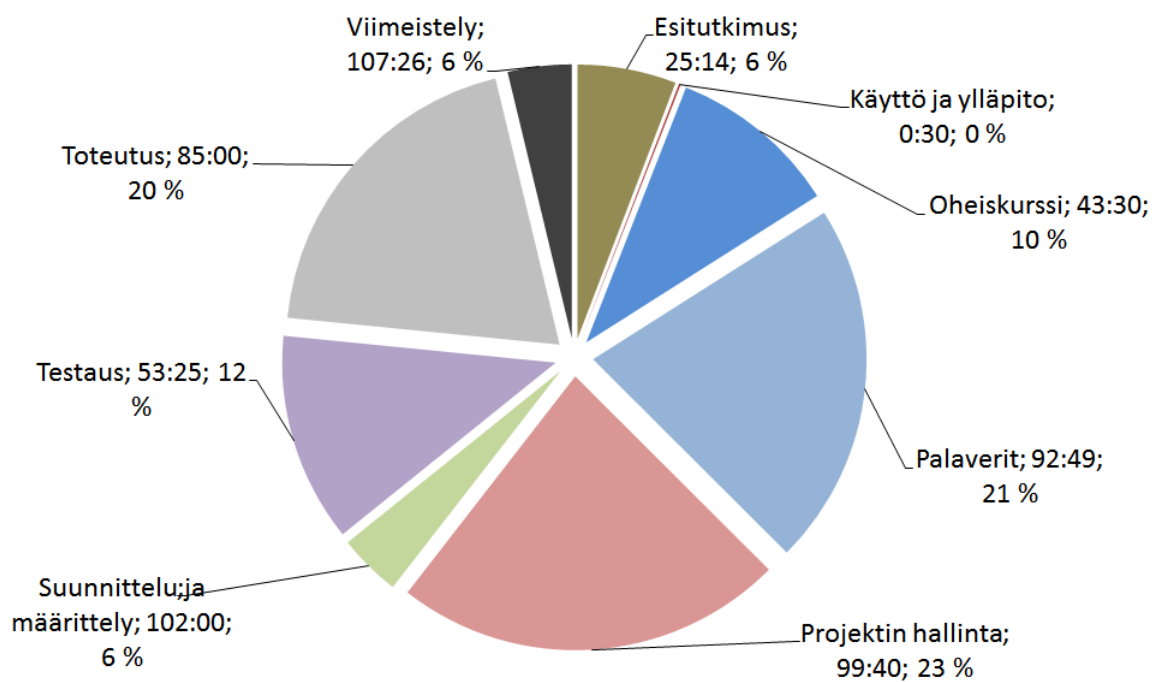
Puupponen vastasi vaatimusmäärittelystä ja palvelinsovelluksesta. Palvelinsovelluksen myötä Puupponen joutui perehtymään käytettyihin työkaluihin, tietojärjestelmän suunnitteluun ja tekemään paljon ohjelmointia, mikä näkyy selvästi tehdyistä työtunneista kuvassa 6.5. Puupponen teki myös paljon lähdekoodin viimeistelyä ja laati palvelinsovelluksen asennusohjeen. Puupposen muita huomattavasti suuremmista työmääristä johtuen tehtäväkokonaisuuksien osuudet jäivät keskimääristä pienemmäksi.



Kuva 6.5: Veli-Mikko Puupposen työtunnit tehtäväkokonaisuuksittain.

6.6 Ilkka Rautiaisen työtunnit tehtäväkokonaisuuksittain

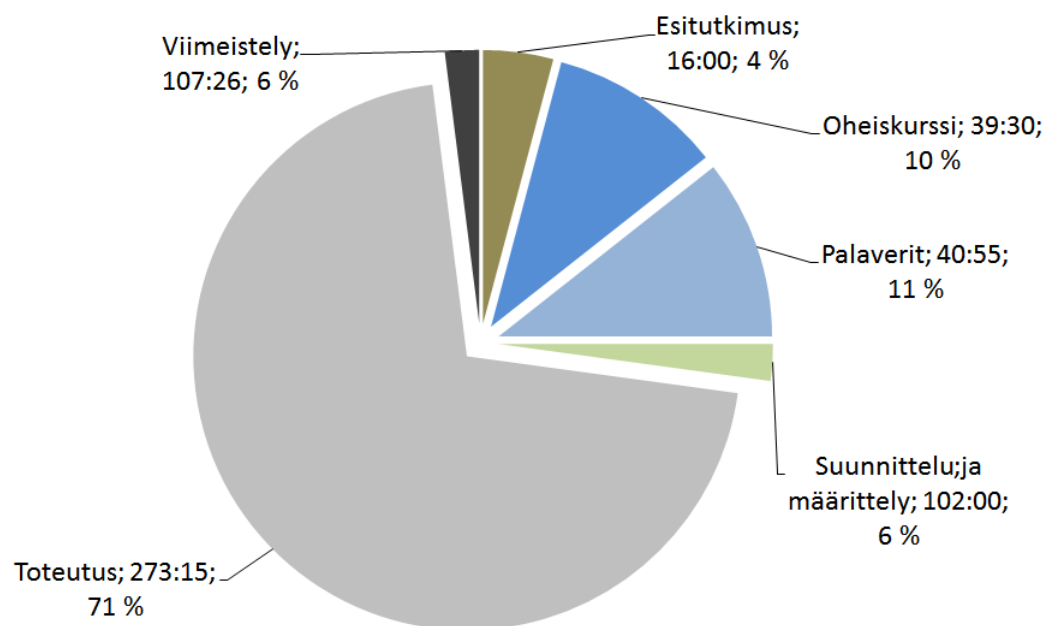
Rautiaisen työtunnit tehtäväkokonaisuuksittain on esitetty kuvassa 6.6. Rautiainen vastasi järjestelmätestauksen ohella sovellusraportista, mikä näkyy projektin hallinnan suuresta osuudesta. Tämän lisäksi Rautiainen toimi myös usein sihteerinä, joten hänelle kertyi paljon tunteja pöytäkirjojen laatimiseen. Rautiainen toteutti myös joitakin palvelimen ja rajapintojen osia.



Kuva 6.6: Ilkka Rautiaisen työtunnit tehtäväkokonaisuuksittain.

6.7 Atte Söderlundin työtunnit tehtäväkokonaisuuksittain

Söderlund vastasi hätäkeskus- ja mobiilisovelluksen toteuttamisesta, mikä näkyy selvästi hänen toteutuksen työtunneistaan kuvassa 6.7. Palaveriinkin käytetty muita ryhmän jäseniä pienempi tuntimäärä selittyy sillä, että Söderlund ei laatinut kuin yhden pöytäkirjan. Söderlund tuurasi projektipäällikköä vain muutaman kerran, eikä hän kirjannut niitä tunteja projektinhallinnan työtunteihin.



Kuva 6.7: Atte Söderlundin työtunnit tehtäväkokonaisuuksittain.

7 Ohjelmistokehitysprosessi ja aikataulu

Luvussa kuvataan projektissa noudatettua prosessia ja aikataulua. Toteutunut aikataulu vastasi rakenteeltaan hyvin suunniteltua. Vaatimusmäärittely ja projektisuunnitelma venyivät aikataulustaan, mutta vaatimusmäärittelyn kuuluukin elää projektin mukana. Alustan kanssa kohdatut tekniset haasteet hidastivat videopuhelun toteuttamista. Tilaajan yllättäen toivoman esittelyvideon suunnittelun ja toteuttamisen sekä kohdattujen teknisten haasteiden seurauksena projektin aikataulu venyi kuukaudella suunnitellusta aikataulusta pelivarakin huomioiden.

7.1 Prosessi

Projektin alussa oli määrittelyn ja suunnittelun vaihe. Siinä suunniteltiin projektin läpivienti, kartoitettiin järjestelmän vaatimukset ja laadittiin vaatimusmäärittely sekä hahmoteltiin sovelluksen käyttöliittymä, rakenne ja arkkitehtuuri.

Ensimmäisessä kehitysvaiheessa toteutettiin mobiilisovelluksen, palvelimen ja hätäkeskuksen käyttöliittymän runko hyvin pienellä määrällä toivotuista toiminnoista. Kun runko oli kasassa ja todettu toimivaksi, alettiin siihen toteuttamaan lisää ominaisuuksia vaatimusmäärittelyn pohjalta kahden viikon vaiheissa. Ohjelmistokehitysvaiheita oli yhteensä viisi. Vaiheen aloittavassa palaverissa esiteltiin sen hetkistä sovellusta ja päätettiin tilaajan kanssa, mitä uusia ominaisuuksia kehitetään seuraavaksi. Jokaiseen vaiheeseen sisältyi ohjelmiston suunnittelua, toteutusta ja testausta. Projektin lopuksi oli tulosten viimeistelyn ja raportoinnin vaihe.

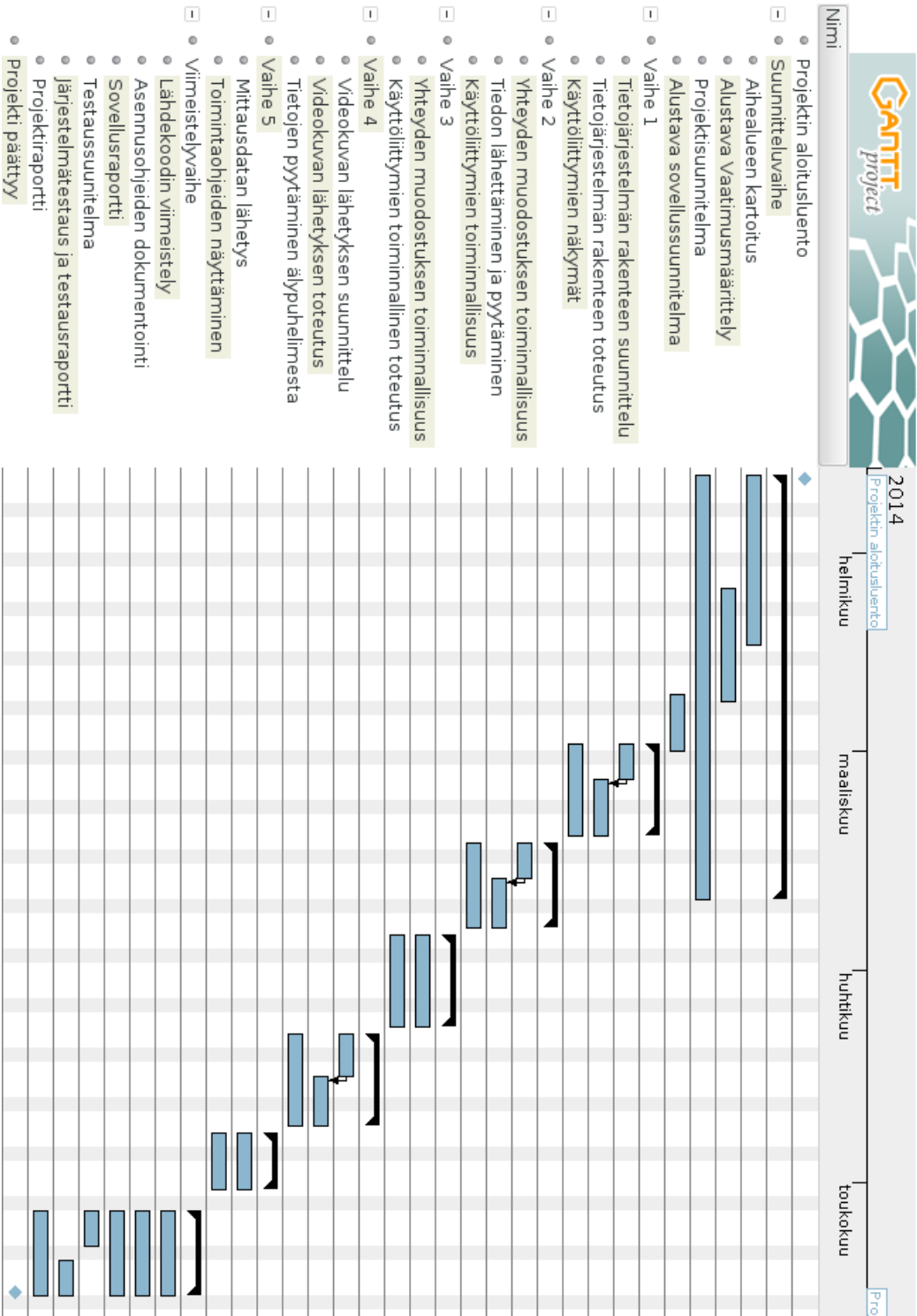
Kehitysvaiheet venyivät suunnitellusta aikatauluistaan, ja ne eivät näkyneet projektiryhmälle kovin selkeästi. Kehitysvaiheet aloitettiin suunnitellun aikataulun mukaisesti, mutta venymisen vuoksi ne sulautuivat yhdeksi kehitysprosessiksi. Tämä ei kuitenkaan haitannut projektin läpivientiä, vaan näin pystyttiin hiomaan ja korjaamaan aikaisemmissa vaiheissa kesken jääneitä ominaisuuksia samalla, kun uusia ominaisuuksia toteutettiin.

7.2 Aikataulu

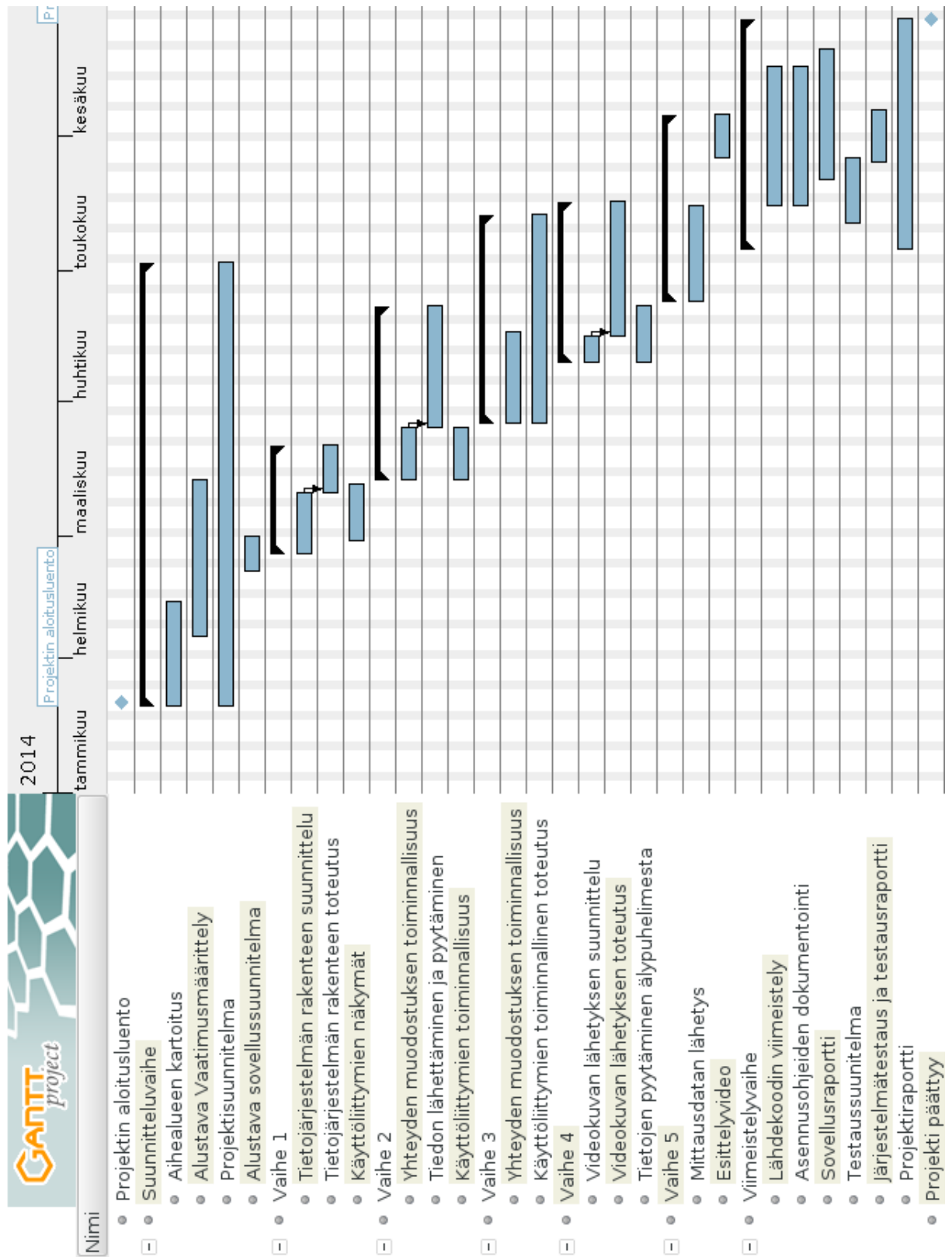
Kuvassa 7.1 on esitelty projektin suunniteltu aikataulu. Projekti alkoi 21.1.2014, ja sen suunniteltiin päättyvän toukokuun puolella välissä, jonka jälkeen oli varattu 2 viikon pelivara. Ensimmäinen yksinkertainen prototyyppi oli valmis 10.4.2014. Tämän jälkeen kahden viikon vaiheissa toteutettiin lisää ominaisuuksia palaverissa sovitun mukaisesti. Toukokuussa ja kesäkuussa ei toteutettu uusia ominaisuuksia, vaan keskityttiin ohjelmiston keskeneräisten ominaisuuksien viimeistelyyn, lähdekoodin viimeistelyyn ja kommentointiin sekä dokumenttien laatimiseen ja viimeistelyyn.

Toteutuneen aikataulun rakenne kuvassa 7.2 vastaa hyvin pitkälti suunniteltua. Toiseksi viimeisessä kehitysvaiheessa kohdattujen ongelmien vuoksi aikataulu venyi kuukauden yli suunnitellun pelivaran. Myös projektisuunnitelman laatiminen venyi kuukauden yli suunnitellun, mikä osaltansa hankaloitti projektin hallintaa. Projektin lopulla tilaajan esittämä vaatimus esittelyvideosta viivästytti aikataulua viikolla, koska sitä ei ollut suunniteltu etukäteen. Projekti päättyi kesäkuun lopussa, eikä alunperin suunnitellusti 16.5.2014. Projektille oli varattu pelivaraa toukokuun loppuun, joten projekti viivästyi siitäkin neljä viikkoa.

Suurin syy projektin viivästymiselle on käytettyjen työtuntien määrä ylitti suunnitellun. Tämä johtui kohdatuista haasteista sekä tietojärjestelmän laajuudesta ja arvioitua suuremmasta työmäärästä. Myös Monosen ja Rautiaisen muut kiireet vaikuttivat siihen, etteivät he pystyneet käyttämään tarpeeksi työtunteja projektin alussa.



Kuva 7.1: Projektin suunniteltu aikataulu.

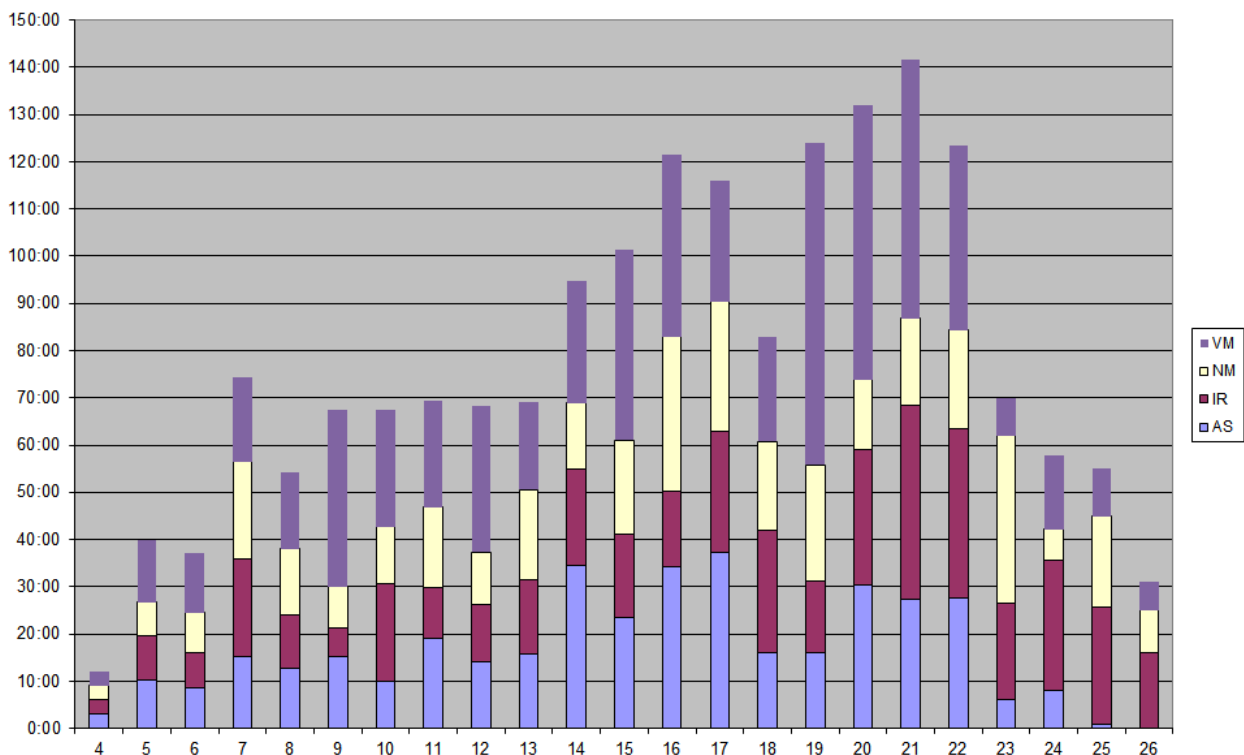


Kuva 7.2: Projektin toteutunut aikataulu.

7.3 Ryhmän työtunnit viikoittain

Kuvassa 7.3 on esitetty ryhmän käyttämien työtuntien jakautuminen eri viikoilla. Suunniteltu sovellusprojektin työtuntimäärä viikkoa kohden oli 14 tuntia jokaista jäsentä kohden, jonka lisäksi oli vielä varattu keskimäärin neljä tuntia oheiskursseihin. Työtunteja ei kertynyt yhdelläkään viikolla tasaisesti ryhmän jäsenten kesken.

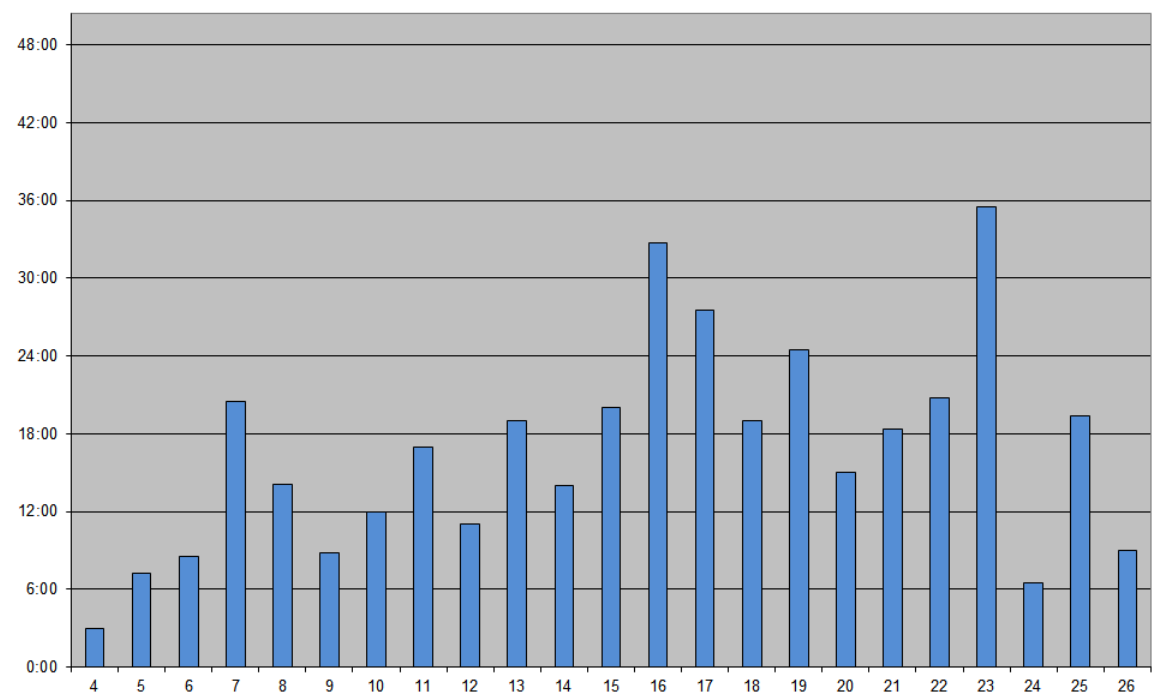
Suunniteltuja työtunteja oli noin 72 tuntia viikossa. Projektin kolmena ensimmäisenä viikkona ryhmä jäi jonkin verran suunnitellun alle. Sen jälkeen pysyttiin melko hyvin suunnitellussa, kunnes kohdattiin ongelmia alustan kanssa, jolloin työtunnit ylittivät reilusti suunnitellun viikottaisen työtuntimäärän. Projektin lopussa jäsenille kertyneiden työtuntien eroa pyrittiin tasaamaan, mikä näkyy Puupposen pienempinä työtunteina kuvassa 7.3.



Kuva 7.3: Ryhmän työtunnit viikoittain.

7.4 Niko Monosen työtunnit viikoittain

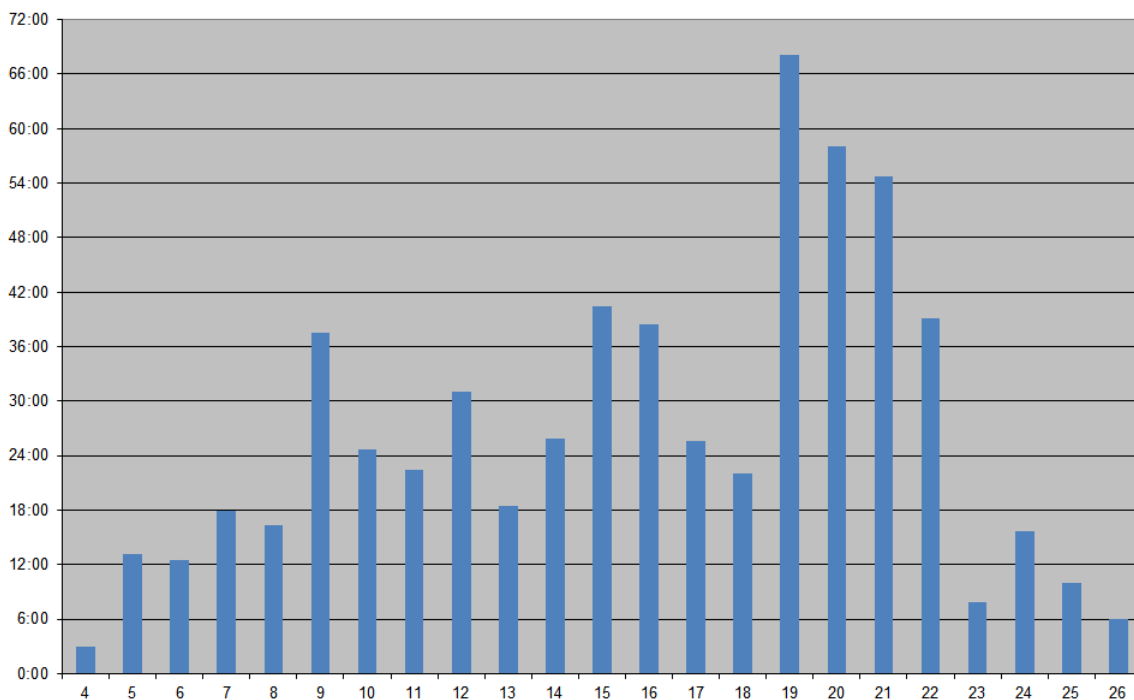
Kuvassa 7.4 näkyy, että Monosen viikoittaiset työtuntimäärät vaihtelivat suuresti. Muut kiireet vähensivät projektiin käytettyjen työtuntien määrää joillakin viikoilla, mutta niitä pyrittiin tasaamaan seuraavien viikkojen aikana. Loppua kohden kasvanut tuntimäärä kertoo muiden kiireiden vähenemisestä, jolloin Mononen on pystynyt panostamaan enemmän projektiin. Mononen oli koko projektin ajan osaaikaisesti ja kesäkuun alusta lähtien täysipäiväisesti töissä, mikä vaikutti työmääriin.



Kuva 7.4: Niko Monosen työtunnit viikoittain.

7.5 Veli-Mikko Puupposen työtunnit viikoittain

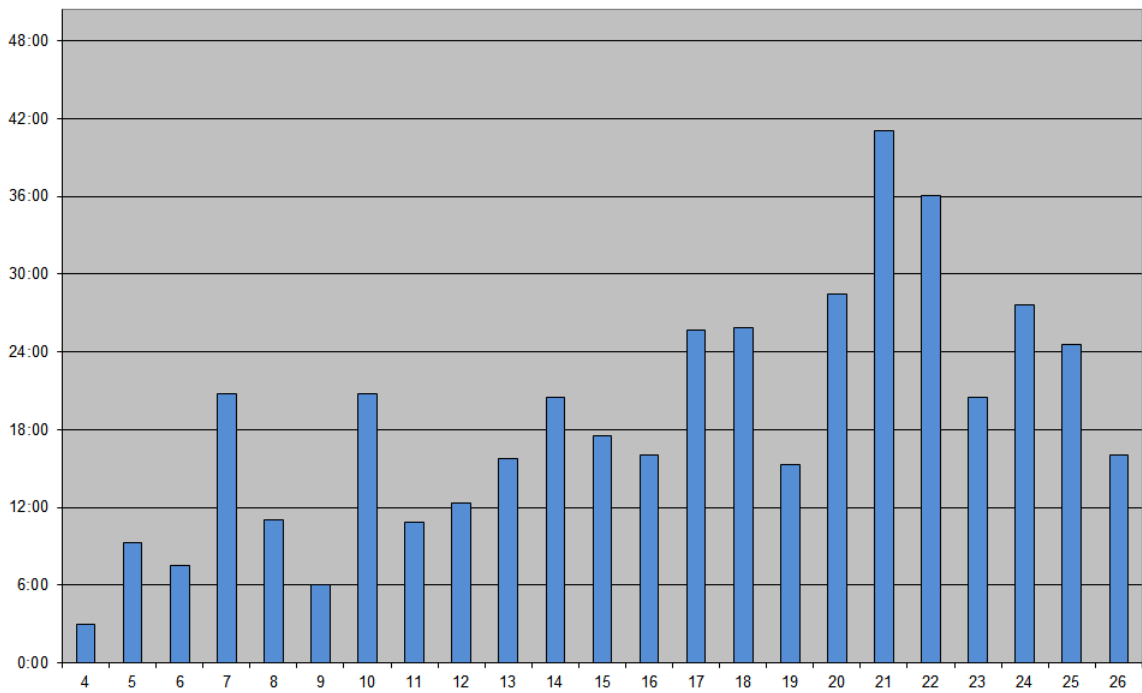
Kuvasta 7.5 näkyy, että Puupposen työtunnit ylittivät suunnitellun 18 tuntia viikossa lähes kaikilla viikoilla. Puupposen panostus vaatimusmäärittelyyn viikolla 9 ja videon siirron ongelmien ratkaisemiseen viikoilla 19–22 näkyy selvästi hänen tehdyistä työtunneista. Kuvasta näkyy myös työtuntien taseus projektin lopussa. Kuvan mittakaava poikkeaa kuvista 7.4, 7.6 ja 7.7.



Kuva 7.5: Veli-Mikko Puupposen työtunnit viikoittain.

7.6 Ilkka Rautiaisen työtunnit viikoittain

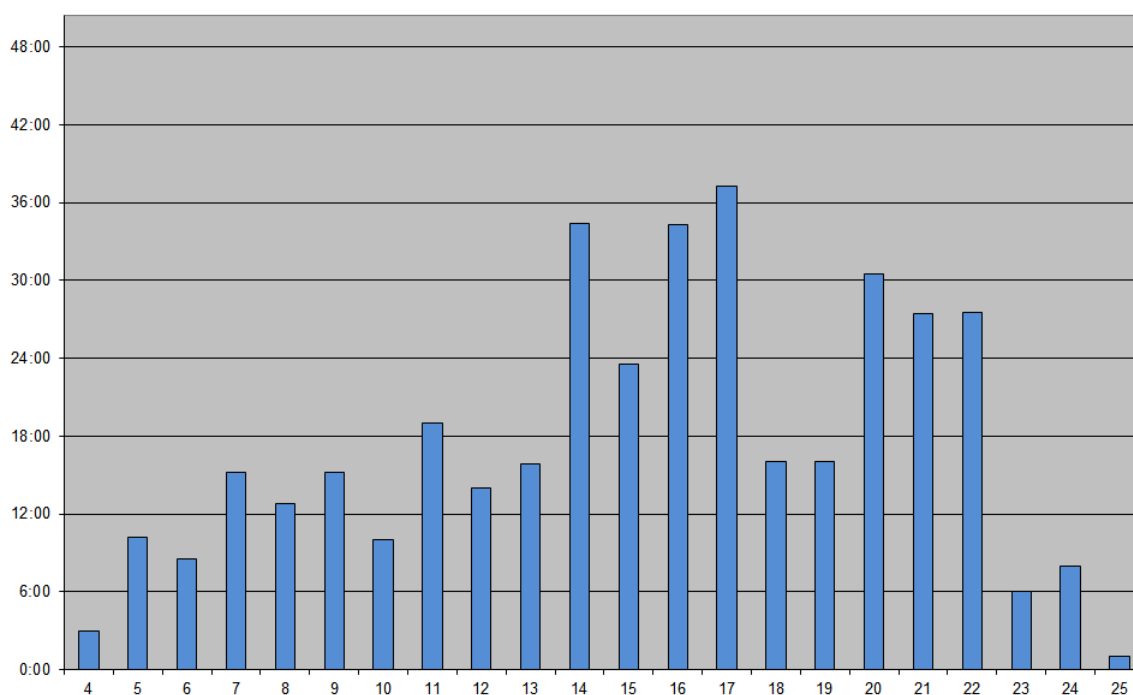
Kuvasta 7.6 näkyy Rautiaisen muiden kurssien vaikutus käytettyihin työtunteihin projektin alussa. Kuvassa näkyy myös Rautiaisen suorittama järjestelmätestaus ja sovellusraportin laatiminen suurina viikkotunteina projektin lopussa.



Kuva 7.6: Ilkka Rautiaisen työtunnit viikoittain.

7.7 Atter Söderlundin työtunnit viikoittain

Kuvasta 7.7 näkyy, että Söderlundilla oli myös muita kiireitä eikä hän panostanut tarpeeksi käyttöliittymien suunnitteluun projektin alussa. Kuvasta näkyy selvästi myös lomien ja vapaiden vaikutus Söderlundin projektiin käyttämiin työtunteihin. Esimerkiksi viikolla 16 oli pääsiäinen, jolloin Söderlund käytti lähes tuplasti työtunteja. Kesäkuussa Söderlund muutti Tampereelle ja aloitti kesätyöt, mikä myös näkyy kuvasta vähäisinä työtunteina.



Kuva 7.7: Atter Söderlundin työtunnit viikoittain.

8 Riskit ja niiden hallinta

Projektin suurimpina riskeinä toteutuivat alustan haasteellisuus ja kehittäjien tietotaidon puutteet. Kyseiset riskit saatiin kuitenkin hallittua projektin aikana, joten ne eivät estäneet projektin läpivientiä, mutta vaikuttivat hieman tuloksiin. Toteutuneista rikseistä johtuneiden ongelmien ratkaisemiseen jouduttiin kuitenkin käyttämään suunniteltua enemmän työtunteja. Luvussa kuvataan projektin ennakoitujen riskien toteutumista, ehkäisemistä ja käsittelyä sekä niiden vaikutusta projektin läpivientiin ja tuloksiin.

8.1 Riskien todennäköisyydet ja haitat

Riskien todennäköisyydet sekä niiden arvioidut ja toteutuneet haittavaikutukset on esitetty taulukossa 8.1. Todennäköisyyttä ja haittavaikutusta arvioitiin asteikolla pieni, keskinkertainen ja suuri.

Riski	Arvioitu todennäköisyys	Arvioitu haittavaikutus	Toteutunut haittavaikutus
Alustan rajoitukset	keskinkertainen	suuri	suuri
Kehittäjien tietotaidon puutteet	keskinkertainen	keskinkertainen	suuri
Tavoitteiden rajaaminen	keskinkertainen	keskinkertainen	pieni
Tiedotuksen puutteet	pieni	keskinkertainen	keskinkertainen
Jäsenten poissaolot	keskinkertainen	pieni	pieni
Kokemattomuus projektihallinnassa	keskinkertainen	pieni	keskinkertainen

Taulukko 8.1: Riskien todennäköisyys ja haittavaikutus.

Suurimmat erot arvioitujen- ja toteutuneiden haittavaikutusten välillä oli kehittäjien tietotaitojen puutteissa ja kokemattomuudessa projektinhallinnassa. Haittavaikutukset tavoitteiden rajaamisessa jäivät odotettua pienemmiksi. Projektin aikana ei ilmennyt ennakoimattomia riskejä.

8.2 Alustan rajoitukset

Projektiryhmä joutui päättämään yhdessä tilaajan edustajien kanssa, mille alustoille tietojärjestelmään kuuluvat sovellukset toteutettiin projektin aikana. Koska alustat ovat hyvin erilaisia, jonkin ominaisuuden toteuttaminen olisi voinut vaatia huomattavasti enemmän resursseja jollakin alustalla. Alustan valinnasta johtuen joitakin tietojärjestelmältä vaadittuja ominaisuuksia oltaisiin voitu joutua rajaamaan projektin ulkopuolelle.

Riskiä ehkäistiin tutustumalla alustavasti etukäteen eri alustoihin. Riskin toteutuksessa oli mahdollista vaihtaa käytettävää alustaa tai rajata joitakin toteutettavia ominaisuuksia projektin ulkopuolelle.

Valittu alusta osoittautui ennakoidusti haasteelliseksi. Tämä johtui WP8-alustan valmiskomponenttien keskeneräisyydestä sekä puutteellisesta ja heikosta dokumentaatiosta. Projektiryhmä joutui useasti päättämään yhdessä tilaajan edustajien ja teknisten ohjaajien kanssa, miten kohdattu ongelma ratkaistaan. Alustaa ei kuitenkaan jouduttu vaihtamaan, ja projektille asetettuja vaatimuksia ei muutettu.

Riskistä johtuen toteutettuun prototyyppiin jouduttiin toteuttamaan paljon komponentteja itse. Näitä komponentteja ei ole optimoitu, eikä testattu kovinkaan perusteellisesti. Nämä komponentit kuitenkin osoittavat tekniikan olevan mahdollista toteuttaa. Prototyypin toteuttaminen toisella alustalla olisi todennäköisesti vaatinut vähemmän työtunteja.

8.3 Kehittäjien tietotaidon puutteet

Ohjelmiston toteutuksessa käytettiin monia eri työkaluja ja tekniikoita, joista projektiryhmän jäsenillä ei ollut aikaisempaa kokemusta. Näin ollen ryhmä ei aina osannut ennakoida mahdollisia ongelmakohtia.

Riskin ehkäisemiseksi ryhmän jäsenet tutustuivat käytettäviin työkaluihin ja tekniikoihin. Ongelmatilanteissa projektiryhmällä oli myös käytettävissään kaksi teknistä ohjaajaa ja asiantuntijoita.

Projektisuunnitelmassa ennakoidusti ohjelmistojen toimintaan tutustumiseen jouduttiin käyttämään paljon aikaa, joka taas vei aikaa itse sovelluksen kehittämisestä. Lisäksi työkalujen ja ohjelmointikielen hyviä ominaisuuksia ei välttämättä heti osattu hyödyntää, vaan asiat tehtiin niin sanotusti "vaikeamman kautta". Lisäksi

WP8-alustan valmiskomponentit olivat keskeneräisiä, sekä niiden dokumentaatio oli puuttellisista ja heikkoa.

Projektiryhmän olisi ehkä pitänyt olla aktiivisemmin yhteydessä teknisiin ohjaajiin liittyen sovelluksen toteutukseen, jotta riskiä toteutuksen tietotaidon puutteissa olisi saatu entisestään ehkäistyä. Joihinkin asioihin edes tekninen ohjaaja ei osannut vastata, vaan asia jouduttiin selvittämään itse käyttämällä siihen paljon työtunteja.

Riskistä johtuen toteutettuun prototyyppiin jouduttiin toteuttamaan paljon komponentteja itse. Näitä komponentteja ei ole optimoitu, eikä testattu kovinkaan perusteellisesti. Nämä komponentit kuitenkin osoittavat tekniikan olevan mahdollista toteuttaa. Riskin toteutumisesta seurasi ryhmän jäsenten arvioitua suuremmat työtunnit ja projektin viivästyminen tavoitteesta.

8.4 Tavoitteiden rajaaminen

Tilaaajan edustajien tavoitteet olivat enimmäkseen hyvin yleisen tason kokonaistavoitteita, joten vaatimusmäärittelyn laatimiseen oli kiinnitettävä paljon huomiota. Ryhmän muodostama kuva ohjelmiston tavoitteista olisi voinut erota tilaaajan toiveista ja tarpeista. Tällöin tietojärjestelmä ei olisi tullut projektin päättyessä sisältämään kaikkia vaatimusmäärittelyssä kuvattuja olennaisimpia ominaisuuksia.

Riskin ehkäisemiseksi projektiryhmä joutui projektin alussa käyttämään paljon aikaa sovittaakseen tilaaajan näkemykset yhtenäiseksi vaatimusmäärittelyksi [9]. Siihen kirjattiin projektissa toteutettavat vaatimukset ja priorisoitiin ne. Ennen kuin ohjelmiston toteutus aloitettiin, projektiryhmällä ja tilaajalla olikin siten melko yhtenäinen kuva siitä, mitä ohjelmiston osia ja toimintoja oli mahdollista saada projektin aikana valmiiksi.

Riski ei juurikaan toteutunut ehkäisyn onnistumisesta johtuen. Tilaaajan edustajat olivat joustavia tavoitteiden suhteen, ja projektille asetettuja tavoitteita ei jouduttu muuttamaan.

8.5 Tiedotuksen puutteet

Säännöllisesti järjestetyissä palaverissa tilaajan edustajat saivat hyvin tietoa projektin tilasta. Palaverien ulkopuolella tuloksista tiedotettiin, mutta tiedotus suorite-
tuista tehtävistä tai projektin etenemisestä oli olematonta, mikä laski projektin läpi-
viennin läpinäkyvyyttä.

Ryhmän sisäisessä tiedotuksessa ei ollut ongelmia. Ryhmä oli usein samassa tilassa työskentelemässä, sekä kaikki muut saatiin hyvin nopeasti tavoitettua sähköpostien ja pikaviestimien avulla. Kommunikaatio sujui ongelmitta, ja tiedotus tuntui toimivan.

Ennakoitu riski tiedotuksen puutteesta toteutui osittain, mutta sen haittavaikutukset jäivät arvioiduksi. Projektiryhmän aktiivisemmalla tiedottamisella olisi ollut mahdollista saada enemmän palautetta tilaajalta. Joissakin tilanteissa tiedotus viivästyi, mikä johtui projektin hallinnan puutteista. Tiedotuksen viivästyminen vaikutti erityisesti aikataulujen sopimiseen.

8.6 Jäsenten poissaolot

Projektin aikana ryhmän jäsenille ei ilmaantunut merkittäviä odottamattomia poissaoloja. Kaikilla ryhmän jäsenillä oli velvoitteita muillakin kursseilla ja työelämässä, mutta mitkään niistä eivät varsinaisesti olleet odottamattomia. Joihinkin toisten kurssien työtehtäviin meni odotettua enemmän aikaa, mutta se ei vaikuttanut projektin läpivientiin.

Riskiä hallittiin projektin aikana siten, että ryhmän jäsenet keskustelivat projektin aikana työtehtävistään. He myös pitivät huolen siitä, että myös muut tiesivät, mitä kukin jäsen on projektin aikana tehnyt. Täten yllättävän poissaolon toteutuessa ryhmän jäsenet olisivat tienneet poissaolijan työtehtävät, ja niitä olisi voinut jakaa muiden jäsenten kesken.

Vaikka ryhmän jäsenet olivat enimmäkseen tietoisia toistensa tekemisistä ja projektia edistävistä toimenpiteistä sekä hidastavista puutteista, ryhmän jäsenet eivät aina toimineet asian hyväksi muistuttamalla tai ottamalla asian hoidettavaksi. Poissaolot vaikuttivat suunniteltuun aikatauluun viivästyttämällä tuloksia ja täten projektia.

8.7 Kokemattomuus projektihallinnassa

Projektiryhmän jäsenillä ei ollut aiempaa kokemusta projektin suunnittelusta, läpiviennistä tai hallinnasta. Myös projektipäällikkö vastasi projektin läpiviennistä ensimmäistä kertaa. Vaikka riski ennakoitiin toteutuvaksi, suoriuduttiin projektin läpiviennissä ilman suurempia ongelmia.

Projektipäällikkö pyrki pitämään itsensä tietoisena siitä, mitä kukin ryhmän jäsen on tehnyt, tekee tällä hetkellä ja aikoo tehdä seuraavaksi. Ryhmän jäsenet kirjasiivat työtunnit projektin käytänteiden mukaisesti. Kaikki jäsenet olivat yleensä tietoisia siitä, missä kehitysvaiheessa projekti eteni. Lisäksi projektiryhmä auttoi projektipäällikköä. Ryhmän jäsenet muistuttivat projektipäällikköä, jos jokin asia uhkasi jäädä huomiotta. Myös vastaava ohjaaja auttoi projektipäällikköä projektin läpiviennissä.

Ryhmän jäsenet ja etenkin projektipäällikkö eivät osanneet arvioida vaadittavia tömääriä riittävällä tarkkuudella. Riskin toteutuminen keskinkertaisena aiheutti projektin viivästymistä suunnitellusta aikataulusta.

9 Jäsenten kokemuksia ja oppimaa

Projekti oli kaikille ryhmän jäsenille ensimmäinen oikea ohjelmistokehitysprojekti, joka sisälsi projektin kaikki kehitysvaiheet aina vaatimusmäärittelystä projektin tulosten luovutukseen. Projekti oli ryhmän jäsenten mielestä opettavainen ja hyödyllinen kokemus. Jäsenet saivat kattavan kuvan projektin läpiviennistä, sekä ymmärsivät, että sovelluksen kehittäminen on kaikkea muutakin kuin pelkkää ohjelmointia. Kaikki ryhmän jäsenet oppivat projektin aikana ryhmätyö- ja palaveritaitoja sekä pääsivät tutustumaan ja suorittamaan heitä itseään kiinnostavia työtehtäviä. Luvuissa 9.1–9.4 projektiryhmän jäsenet kuvaavat kokemuksiaan ja oppimiaan asioita.

9.1 Niko Monosen kokemuksia ja oppimaa

Projektin kohdealue vaikutti todella haastavalta ja mielenkiintoiselta. Tietojärjestelmäkokonaisuudet ovat aina olleet sydäntäni lähellä, ja päätin ottaa projektipäällikön tehtävän vastuulleni.

Sain selkää ohjausta ja tukea muilta ryhmän jäseniltä. Nämä asiat helpottivat huomattavasti työtäni ja tehostivat oppimistani merkittävästi. Ryhmän jäsenet tekivät myös paljon pyytämättäni, kun en osannut ottaa kaikkea huomioon. Projektipäällikön tehtävät osoittautuivat melko haastaviksi ja yllättävän paljon aikaa vieviksi.

Erilaisten dokumenttien kirjoittaminen tuntui turhauttavalta, mutta ymmärrän kyllä, että ilman suunnittelua ja raportointia projektin läpiviennistä ei tulisi mitään. Jouduin kuitenkin käyttämään huomattavan paljon aikaa niiden laatimiseen, jotta niistä tuli selkeitä, ymmärrettäviä ja kieliasultaan riittävän oikeita. Sain paljon palautetta dokumenteista myös vastaavalta ohjaajalta, mikä vaikutti dokumenttien laatuun.

Projektin aikana pyrin pitämään itseäni koko ajan tietoisena siitä, mitä ryhmän jäsenet tekevät ja mitä he tulevat tekemään. Mielestäni onnistuin tässä hyvin. Olisin kaivannut ryhmän jäseniltä hieman enemmän oma-aloitteisuutta tekemisiensä raportoinnissa. Tuntui jotenkin turhauttavalta kysellä jatkuvasti, että mitä olet tehnyt ja mitä teet seuraavaksi.

Projektipäällikön tehtävät kuitenkin veivät työaikaani varsinaiselta ohjelmoinnilta, ja se harmitti minua paljon. Projektin aikana toteutin vain muutaman pienen toimin-

nallisuuden. Tämä ei kuitenkaan varsinaisesti haittaa, sillä sain tärkeää kokemusta projektin johtamisesta.

Projektiryhmä oli erinomainen. Kaikki ymmärsivät vastuunsa ja tekivät töitä todella tunnollisesti. Välillä tuntui, että muilla jäsenillä oli paljon aikaa tehdä projektia, enkä itse pysynyt lainkaan mukana työtunneissa. Tämä johtui siitä, että olin puoli-päiväisenä töissä ja minulla oli muitakin työläitä kursseja. Projektiin käytössä oleva työaikani oli yleensä vasta klo 16 jälkeen, mikä selvästi haittasi yhteistoimintaa muun ryhmän kanssa.

Kokonaisuutena sovellusprojekti oli positiivinen kokemus, vaikka toukokuussa minäsin uupua kaikesta itselleni hankkimasta työmäärästä. Koen saaneeni todella tärkeää kokemusta projektimuotoisesta työtavasta tällä kurssilla.

9.2 Veli-Mikko Puupposen kokemuksia ja oppimaa

Jo ennen projektin alkua tiesin muilta kuulemieni kokemusten perusteella odottaa työpanokseltaan ja ajankäytöltään vaativaa kurssia. Projektin käynnistyttyä tämä osoittautuikin todeksi, ja päädyin tekemään töitä tahdilla, jolla kevätlukukaudelle valitsemalleni kurssille jäi hädin tuskin aikaa. Toisaalta samalla positiivista oli, että työtä päästiin tekemään kohtuullisen hyvällä teholla suhteellisen pian projektin alettua, vaikka aihealue ei ollut entuudestaan lainkaan tuttu.

Projektin alettua otin yhdeksi vastuualueekseni vaatimusmäärittelyn. Sen kokoomista helpottivat selvästi samoihin aikoihin suorittamani vaatimusmäärittelyn kurssi ja tilaajan hyvä käsitys tarpeellisista ominaisuuksista. Uusia ideoita kertyi myös projektin kuluessa jatkuvasti lisää, ja vaatimusmäärittely kasvoi projektin loppuun mennessä useiden satojen vaatimusten mittaiseksi. Tilaajan selkeän näkemyksen ansiosta projektin kannalta olennaisimmat vaatimukset saatiin kuitenkin kirjattua jo aivan projektin alkuvaiheessa. Vaatimusmäärittelyn jatkuva kehittäminen palaverissa käytyjen keskustelujen ja saadun palautteen pohjalta antoi puolestaan hyvää käytännön kokemusta vaatimusmäärittelyyn liittyvien tehtävien laajasta kokonaisuudesta.

Projektissa käyttämäni C#-ohjelmointikieli oli minulle entuudestaan tuttu, mutta yli vuoden kestäneen tauon jälkeen oli hyödyllistä oppia siihen tulleista uusista ominaisuuksista. Kehittäessäni projektin palvelinohjelmistoa sain kokonaan uutta tietoa erilaisista verkkoliikennekirjastoista ja IIS-palvelimesta. Mielenkiintoista oli myös

päästä kokeilemaan ohjelmointia Windows Phone 8 -käyttöjärjestelmälle aikaisemman Android-ohjelmointikokemuksen pohjalta. Lisäksi toteutin projektin järjestelmään monia kuvan- ja äänensiirron ratkaisuja, jotka antoivat reaaliaikaisen viestinnän kysymyksistä paljon hyödyllistä kokemusta.

Projektimuotoisen työtavan lisäksi epäilemättä tärkein projektin aikana ja sen oheiskursseilla saamani kokemus oli orientoituminen ryhmätyöhön ennalta tuntemattomien jäsenten kanssa. Vaikka ryhmätyökokemusta sinänsä oli kertynyt useammalta ohjelmointikurssilta, vaati sovellusprojekti laajuutensa ja pituutensa takia selvästi tiiviimpää yhteistyötä ja yhteisiin tavoitteisiin sitoutumista.

9.3 Ilkka Rautiaisen kokemuksia ja oppimaa

Olin kuullut Sovellusprojekti-kurssista ja sen työläydestä paljon juttuja. Toisaalta moni oli kehunut kurssia, ja odotin aiemmin minulle vähiin jäänyttä projektimuotoista työtapaa mielenkiinnolla. Tiesin myös, että minulla oli kevätlukukaudella muutama muukin kiinnostava ja työläs kurssi suoritettavana. Odotettavissa oli siis kiireinen kevät.

Ohjelmoinnissa käytetty C#-kieli oli minulle perustasolla tuttua, mutta lähes kaikki muu projektissa vastaan tullut tekniikka olikin sitten vierasta. Onneksi projektiryhmästä löytyi taustatiedoiltaan sopivampia ohjelmoijia, jotka hahmottivat kokonaisuuden ja osasivat arvioida siihen parhaiten soveltuvat tekniikat. Projektin alkupuolella ajankäyttöä rajoitti välillä liiankin voimakkaasti toinen kurssi, johon halusin panostaa.

Projektin alkupuolella käytin melko paljon aikaa ymmärtääkseni muiden tekemiä ohjelmointiratkaisuja, ja yritin löytää itselleni joitakin ohjelmointiin liittyviä tehtäviä. Lopulta kirjoittamani ohjelmakoodin määrä on melko vähäinen. Projektin keskivaiheilla pyrinkin siirtämään työni painopistettä dokumentointiin ja muihin tehtäviin, kuten järjestelmätestaukseen. Kirjoitin projektin aikana monta pöytäkirjaa, muistioita ja muita dokumentteja. Sainkin mielestäni projektin aikana parannettua esimerkiksi pöytäkirjojen laadinnassa vaadittavaa rutiininomaista kirjoitustaitoa. Tekstien hiomisessa auttoi vastaavan ohjaajan ja muiden antama rakentava palaute. Sovellusraportin kirjoittamisessa pääsin vielä tutustumaan $\text{\LaTeX}$ -ladontajärjestelmään, joka oli pääasiassa positiivinen kokemus.

Projektimuotoisuudesta muistuttivat lähinnä säännölliset palaverit ja lähes kaiken

tarkka dokumentointi. Muuten projekti eteni mielestäni melko itseohjautuvasti, ja tiedotus ryhmän sisällä toimi hyvin, vaikka projektihuoneessa olivat paikalla kaikki yhtä aikaa vain harvoin.

Lopulta oli mielestäni hieman yllättävääkin, että projektille asetetut tavoitteet täyttyivät näinkin hyvin. Alussa eivät edes projektin tavoitteet olleet tarkasti selvillä, ja projekti osoittautui hyvin laajaksi. Toisaalta täytyy huomata, että alussa asetetut työtuntitavoitteet ylittyivät, ja jakautuivat osin epätasaisesti projektiryhmän jäsenten kesken. Projekti oli kuitenkin kokonaisuutena positiivinen kokemus, ja projektiryhmä toimi keskenään hyvässä hengessä ilman ongelmia.

9.4 Atte Söderlundin kokemuksia ja oppimaa

En tiennyt kurssista juuri mitään, ennen kuin pääsin kurssille jonkun ensin peruttua osallistumisensa. Minulla siis oli ryhmästä selvästi vähiten kokemusta ja opintopisteitä, mutta varmasti eniten kokemusta Windows Phone -alustasta. Varmaan siksi valitsimmekin Windows Phonen.

Kuitenkin opin alustastakin uutta, koska en ole ennen käyttänyt SignalR:ää enkä WCF:ää. Lisäksi tekniseltä ohjaajalta tuli erilaisia auttavia neuvoja, joista on jatkossakin varmasti apua. WPF:stä en oikeastaan oppinut mitään uutta, koska projektimme ei panostanut hätäkeskussovellukseen niin paljon.

Eniten varmasti opin itse projektitoiminnasta eli palavereista ja dokumentoinnista. En ollut aikaisemmin ollut yhdessäkään palaverissa saati sitten laatinut pöytäkirjoja tai muistioita. Olin projektissa varapäälikkönä, mutta en kertaakaan oikein joutunut tekemään mitään projektipäälikön tehtäviä.

Muut kurssit eivät olleen alkuprojektista kovinkaan paljon tiellä, mutta loppusuoralla tentit veivät vähän enemmän aikaa. Kuitenkaan en mielestäni panostanut tarpeeksi alussa suunnitteluun, ja tunteja ei alussa oikein kertynyt. Sitten, kun pääsin toteuttamaan sovelluksia käytännössä, sain loppujen lopuksi kuitenkin melko paljon aikaa käytettyä ja tavoite hieman ylittyikin. Ohjeilmoin käyttöliittymät ja melko paljon toiminnallisuutta niihin lähes kokonaan itse.

10 Yhteenveto

Hälyri-projekti suoritti toteutettavuuskartoituksen ja kehitti hätäilmoitusjärjestelmän prototyypin Sovellusprojekti-kurssilla keväällä 2014. Projektin tulokset luovutettiin tilaajalle kesäkuun lopussa.

Projektin suurimpina riskeinä toteutuivat alustan haasteellisuus ja kehittäjien tietotaidon puutteet. Kyseiset riskit saatiin kuitenkin hallittua projektin aikana, joten ne eivät estäneet projektin läpivientiä, mutta vaikuttivat hieman tuloksiin. Toteutuneista riskeistä johtuneiden ongelmien ratkaisemiseen jouduttiin kuitenkin käyttämään suunniteltua enemmän työtunteja.

Projektissa kehitetyn prototyypin pohjalta tullaan kehittämään tietojärjestelmästä pilottiversio, joten projektissa laadittuja dokumentteja tullaan hyödyntämään myöhemmin. Kehitettävä prototyyppi oli laaja ja monimutkainen, joten tietojärjestelmän sovellusten ominaisuuksien priorisointiin oli kiinnitettävä paljon huomiota.

Jäsenten työtuntien toteuma ylitti huomattavasti suunnitellut. Jokainen jäsen teki projektissa reilut 390 tuntia töitä, joten projektiryhmän kokonaispanokseksi muodostui noin 1780 tuntia. Projektin läpivienti viivästyi pelivarastakin neljällä viikolla.

Toteutunut työtuntien prosentuaalinen jako tehtäväkokonaisuuksittain vastasi lähes suunniteltua, vaikkakin työmäärä ylittyi noin 500 tuntia. Työtuntien jako jäsenten kesken ei koskaan ollut tasaista, vaan ryhmän jäsenten muut kiireet näkyivät työpanoksissa. Puupposelle kertyi noin 200 tuntia enemmän työtunteja muihin ryhmän jäseniin verrattuna kohdattujen haasteiden ja Puupposen oman panostuksensa vuoksi. Ryhmän muiden jäsenten työmäärä jakautui melko tasaisesti.

Vaatusmäärittelyn laatimisessa ja prototyypin kehittämisessä projektiryhmä onnistui suunnitellusti. Vaikka tietojärjestelmän prototyyppi ei vielä sisällä kaikkia vaadittuja ominaisuuksia, tilaajan edustajat voivat käyttää sitä edistääkseen hanketta. Projektin aihealue ja käytetyt kehitysvälineet olivat vaativia, sekä kehitettävä tietojärjestelmä oli hyvin laaja. Projektin läpivienti onnistui erinomaisesti huomoiden, että se oli jäsenille ensimmäinen laajempi ohjelmistokehitysprojekti.

Tietojärjestelmän toteutuksen kokemuksen lisäksi sovellusprojekti antoi projektiryhmän jäsenille kattavan käsityksen ohjelmistoprojektissa työskentelystä sekä sen vaatimuksista ja työtavoista. Ryhmän jäsenet oppivat asioita käytännön tekemisen kautta sekä sovelsivat aikaisemmilta kursseilta oppimaansa teorian tietoja ja taitoja.

11 Lähteet

- [1] Petri Heinonen, "Ajankäytönseurantasovellus", saatavilla Excel-muodossa <URL: <http://appro.mit.jyu.fi/tools/ajankaytto/ajankaytonseuranta.xls>>, Jyväskylän yliopisto, informaatioteknologian tiedekunta, viitattu 23.2.2014.
- [2] Tapio Keränen, Toni Salminen, Jari Salokangas ja Lauri Satokangas, "Paatti-projekti, projektiraportti", saatavilla PDF-muodossa <URL: <http://sovellusprojektit.it.jyu.fi/paatti/>>, Jyväskylän yliopisto, tietotekniikan laitos, 2012.
- [3] Jaana Kuula ja Olli Kauppinen, "Sapporo-raportti", saatavilla WWW-osoitteessa <URL: <https://www.jyu.fi/it/tutkimus-uusi/SAPPORO.pdf>>, Jyväskylän yliopisto, Informaatioteknologian tiedekunta, 2014.
- [4] Tomi Karppinen, "ComTest for C#", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/comtest/wiki/ComTestCsharp>>, Jyväskylän yliopisto, tietotekniikan laitos, 14.2.2014.
- [5] Microsoft, "Windows Phone SDK", saatavilla WWW-osoitteessa <URL: <http://dev.windowsphone.com/en-us/>>, Microsoft, 2014.
- [6] Microsoft, "C# Programming Guide", saatavilla WWW-osoitteessa <URL: <http://msdn.microsoft.com/en-us/library/67ef8sbd.aspx>>, Microsoft, 2014.
- [7] Niko Mononen, Veli-Mikko Puupponen, Ilkka Rautiainen ja Atte Söderlund, "Hälyri-projekti, esittelyvideo", saatavilla WWW-osoitteessa <URL: https://sovellusprojektit.it.jyu.fi/halyri/projektikansio/Halyri_Esittelyvideo_1.0.0.mp4>, Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [8] Niko Mononen, Veli-Mikko Puupponen, Ilkka Rautiainen ja Atte Söderlund, "Hälyri-projektin luokkadokumentaatio", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/sovproj/wiki/Halyri/Luokkadokumentaatio>>, Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [9] Niko Mononen, Veli-Mikko Puupponen, Ilkka Rautiainen ja Atte Söderlund, "Hätäilmoitustietojärjestelmän vaatimusmäärittely", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/sovproj/wiki/>>

Hälyri/Vaatimusmaarittely>, Jyväskylän yliopisto, tietotekniikan laitos, 2014.

- [10] Niko Mononen "Hälyri-projekti, projektisuunnitelma", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/sovproj/wiki/Hälyri/Projektisuunnitelma>>, Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [11] Veli-Mikko Puupponen "Hälyri-projekti, järjestelmätestausraportti", saatavilla WWW-osoitteessa <URL: https://trac.cc.jyu.fi/projects/sovproj/attachment/wiki/Hälyri/Testausraportit/HälyriJarjestelmatestausraportti_VM1.docx>, Jyväskylän yliopisto, tietotekniikan laitos, 27.5.2014.
- [12] Ilkka Rautiainen ja Veli-Mikko Puupponen "Hälyri-projekti, sovellusraportti", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/sovproj/wiki/Hälyri/Sovellusraportti>>, Jyväskylän yliopisto, tietotekniikan laitos, 2014.
- [13] Ilkka Rautiainen "Hälyri-projekti, järjestelmätestausraportti", saatavilla WWW-osoitteessa <URL: https://trac.cc.jyu.fi/projects/sovproj/attachment/wiki/Hälyri/Testausraportit/HälyriJarjestelmatestausraportti_IR_01.pdf>, Jyväskylän yliopisto, tietotekniikan laitos, 23.5.2014.
- [14] Ilkka Rautiainen "Hälyri-projekti, järjestelmätestausraportti", saatavilla WWW-osoitteessa <URL: https://trac.cc.jyu.fi/projects/sovproj/attachment/wiki/Hälyri/Testausraportit/HälyriJarjestelmatestausraportti_IR2.pdf>, Jyväskylän yliopisto, tietotekniikan laitos, 28.5.2014.
- [15] Ilkka Rautiainen "Hälyri-projekti, järjestelmätestausraportti", saatavilla WWW-osoitteessa <URL: https://trac.cc.jyu.fi/projects/sovproj/attachment/wiki/Hälyri/Testausraportit/HälyriJarjestelmatestausraportti_IR3.pdf>, Jyväskylän yliopisto, tietotekniikan laitos, 14.6.2014.
- [16] Ilkka Rautiainen "Hälyri-projekti, järjestelmätestausraportti", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/sovproj/attachment/wiki/Hälyri/Testausraportit/>>

HalyriJarjestelmatestausraportti_IR4.pdf>, Jyväskylän yliopisto, tietotekniikan laitos, 17.6.2014.

- [17] Jukka-Pekka Santanen, "Palaute koekäytöstä", saatavilla WWW-osoitteessa <URL: <https://trac.cc.jyu.fi/projects/sovproj/wiki/Halyri/Koekaytto>>, Jyväskylän yliopisto, tietotekniikan laitos, 16.5.2014.
- [18] Jukka-Pekka Santanen, "Tietotekniikan Sovellusprojektien ohje", saatavilla WWW-osoitteessa <URL: <http://www.mit.jyu.fi/opetus/sovellusprojektit/projohje.html>>, Jyväskylän yliopisto, tietotekniikan laitos, 29.1.2013.