

## **Projektisuunnitelma**

Ohjelmistotuotantoprojektin tietojärjestelmä — OhtuTie

Helsinki 12.7.2004

Ohjelmistotuotantoprojekti

HELSINGIN YLIOPISTO

Tietojenkäsittelytieteen laitos

**Kurssi**

581260 Ohjelmistotuotantoprojekti (6 ov)

**Projektiryhmä**

Janne Nevalainen  
Jyrki Kankaanpää  
Sinikka Loikkanen  
Esa-Matti Miettinen  
Petteri Nurmi  
Max Österman

**Asiakas**

Turjo Tuohiniemi

**Vastuuhenkilö**

Juha Taina

**Kotisivu**

<http://www.cs.helsinki.fi/group/otie/>

**Versiohistoria**

Versio	Päiväys	Tehdyt muutokset
0.1	21.5.2004	Ensimmäinen versio
1.0	28.5.2004	Suunnitelma valmis
1.0	24.6.2004	Aikataulujen päivitys
1.0	02.7.2004	Aikataulujen päivitys

# Sisältö

<b>1</b>	<b>Johdanto</b>	<b>1</b>
1.1	Projektin tarkoitus . . . . .	1
1.2	Projektin kattavuus . . . . .	1
<b>2</b>	<b>Projektin organisaatio ja resurssit</b>	<b>1</b>
2.1	Projektin yhteystiedot . . . . .	1
2.2	Projektin jäsenet ja työnjako . . . . .	2
2.3	Resurssit . . . . .	3
<b>3</b>	<b>Ohjelmisto</b>	<b>3</b>
3.1	Ohjelmiston kuvaus . . . . .	3
3.2	Rakenne . . . . .	3
3.3	Kokoarvio . . . . .	3
<b>4</b>	<b>Projektin eteneminen</b>	<b>6</b>
4.1	Työvaiheet . . . . .	6
4.2	Aikataulu . . . . .	7
<b>5</b>	<b>Työskentelytavat ja viestintä</b>	<b>7</b>
5.1	Kokoukset . . . . .	7
5.1.1	Seurantakokoukset . . . . .	8
5.1.2	Hyväksymiskokoukset . . . . .	8
5.1.3	Tarkastuskokous . . . . .	8
5.2	Työskentelyajat ja -paikat . . . . .	8
<b>6</b>	<b>Dokumentointi ja raportointi</b>	<b>9</b>
6.1	Yleismenettelyt . . . . .	9
6.2	Tuotettavat dokumentit . . . . .	9
<b>7</b>	<b>Ohjaus ja seuranta</b>	<b>10</b>
7.1	Edistymisen seuranta . . . . .	10
7.2	Projektisuunnitelman ylläpito . . . . .	10
7.3	Tulosten hyväksyminen . . . . .	11
7.4	Muutosten, virheiden ja ongelmien hallinta . . . . .	11

	ii
7.5 Laadun valvonta . . . . .	11
<b>8 Riskianalyysi</b>	<b>11</b>
8.1 Eriävät näkemykset toteutettavasta järjestelmästä . . . . .	12
8.2 Projektiryhmään liittyvät riskit . . . . .	12
8.3 Projektin hallintaan liittyvät riskit . . . . .	12
8.4 Tekniikkaan liittyvät riskit . . . . .	12
8.5 Tuotteeseen ja asiakkaaseen liittyvät riskit . . . . .	13

# 1 Johdanto

Oheisessa projektisuunnitelmassa kuvataan OhtuTie-ohjelmistotuotantoprojektin tarkoitus sekä itse projekti. Ohjelmistotuotantoprojektin tietojärjestelmä – OhtuTie – kuuluu Helsingin yliopiston Tietojenkäsittelytieteen laitoksen kuuden opintoviikon laajuiseen Ohjelmistotuotantoprojekti-kurssiin.

## 1.1 Projektin tarkoitus

Projektin tavoitteena on tuottaa ohjelmisto ohjelmistotuotantoprojekteista mitattujen tietojen tallentamiseen ja katselemiseen. Tulevat ohjelmistotuotantoprojektien opiskelijat voivat käyttää ohjelmistoa oman projektinsa etukäteissuunnittelun tukena. Lisäksi opiskelijat lisäävät järjestelmään omaa projektiaan koskevat tiedot. Ohjelmistotuotantoprojektien vastuuhenkilöille ja ohjaajille ohjelmisto antaa entistä paremman ja tarkemman mahdollisuuden projektien etenemisen seurantaan. Lisäksi tutkijat voivat hakea järjestelmästä erilaisia tilastotietoja.

## 1.2 Projektin kattavuus

Projektin kesto on kahdeksan viikkoa. Projekti alkoi 17.5.2004 ja sen on tarkoitus päättyä 9.7.2004. Ehdoton viimeinen takaraja projektin päättymiselle on 16.7.2004.

Projektin aikana tuotetaan ohjelmisto, johon toteutetaan aikajakson puitteissa määrittelydokumentin mukaiset ja siinä priorisoidut toiminnot.

# 2 Projektin organisaatio ja resurssit

Luvussa kuvataan organisaatioon kuuluvat henkilöt ja heidän suhteensa projektiin sekä projektin käytössä olevia resursseja.

## 2.1 Projektin yhteystiedot

**Ryhmän kotisivu:**

<http://www.cs.helsinki.fi/group/otie/>

**Ryhmän hakemisto:**

</home/group/otie/>

**Ryhmän sähköpostilista:**

[ohtuv04-otie-list@cs.helsinki.fi](mailto:ohtuv04-otie-list@cs.helsinki.fi)

## 2.2 Projektin jäsenet ja työnjako

Projektiryhmä koostuu kuudesta henkilöstä. Ryhmän jäsenet ja heidän vastuualueensa on esitetty taulukossa 1.

Taulukko 1: Projektiryhmän jäsenet ja heidän päävastuualueensa

Nimi	Vastuualue
Janne Nevalainen	projektipäällikkö
Jyrki Kankaanpää	dokumenttivastaava
Sinikka Loikkanen	www-vastaava
Esa-Matti Miettinen	testausvastaava
Petteri Nurmi	koodaus- ja ympäristövastaava
Max Österman	sihteeri

**Projektipäällikkö** Janne Nevalainen vastaa projektiryhmän johtamisesta. Hän laatii projektin alussa projektisuunnitelman ja seuraa projektin etenemistä suunnitelman mukaisesti. Projektipäällikkö raportoi projektin vastuuhenkilöille projektin edistymisestä. Projektipäällikkö huolehtii myös asiakkaan ja projektiryhmän välisestä kommunikaatiosta.

**Sihteeri** Max Österman vastaa kokousmuistioiden ja pöytäkirjojen laadinnasta sekä erilaisten dokumenttien ja tiedonantojen välittämisestä kokouksiin osallistuville henkilöille.

**Dokumenttivastaava** Jyrki Kankaanpää huolehtii projektin dokumentaation saattamisesta sisällöllisesti kattavaan ja ulkoisesti siistiin muotoon. Hän luo ryhmän jäsenille yhteäisen säännösten materiaalin kirjoittamisesta ja tallentamisesta projektin aikana. Lisäksi hän vastaa dokumenttien ulkoasun viimeistelystä ja dokumenttien kokoamisesta.

**Ulkoasuvastaava** Sinikka Loikkanen vastaa projektin kotisivujen ylläpidosta ja suunnittelusta sekä tuotettavan sovelluksen käyttöliittymän ulkoasusta. Hän osallistuu myös dokumenttien ulkoasun viimeistelyyn.

**Testausvastaava** Esa-Matti Miettinen vastaa testauksesta, testaussuunnitelmasta ja valvoo suunnitelman noudattamista.

**Koodaus- ja ympäristövastaava** Petteri Nurmi vastaa arkkitehtuurisuunnittelusta, ohjelmointityöstä ja ympäristöstä. Hän vastaa myös siitä, että ryhmän tuottama ohjelmistokoodi on selkeää, kommentoitua ja hyvin suunniteltua.

Projektiryhmän ohjaajana toimii Marianne Korpela ja projektin vastuuhenkilönä tietojenkäsittelytieteen laitokselta Juha Taina. Projektin asiakas Turjo Tuohiniemi on myös tietojenkäsittelytieteen laitokselta.

## 2.3 Resurssit

Projektiryhmän käytössä on tietojenkäsittelytieteen laitoksen atk-palvelut, joihin kuuluu ryhmähakemisto, sähköpostilista sekä cvs-versionhallintajärjestelmä. Projektiryhmä käyttää tietojenkäsittelytieteen laitoksen saleja ja tietokoneita yleisesti sekä erityisesti projektiryhmän käyttöön varattua salia C455. Tietojenkäsittelytieteen laitoksen muuton vuoksi projektiryhmä siirtyy tarvittaessa käyttämään Helsingin yliopiston oppimiskeskus Alexandrian ryhmätyötiloja.

# 3 Ohjelmisto

Luvussa kuvataan pääpiirteissään projektin tuottama ohjelmisto ja sen rakenne. Lisäksi ohjelmiston kokoa arvioidaan rakenteen perusteella tehdyllä laskelmalla.

## 3.1 Ohjelmiston kuvaus

Projektiryhmän tuottama ohjelmisto on tarkoitettu Ohjelmistotuotantoprojektien tietojärjestelmäksi, jonka avulla opiskelijat voivat suunnitella oman projektinsa toteutusta. He myös syöttävät järjestelmään omaan projektiinsa liittyviä tietoja. Projektien ohjaajat ja vastuuhenkilöt voivat seurata ohjelmiston avulla projektien etenemistä. Ohjelmiston avulla kerätyt mittaustulokset tulevat myös tutkimuskäyttöön. Tutkijoille ohjelmisto tarjoaa kuitenkin vain liittymän mittaustietojen visualisointiin.

## 3.2 Rakenne

Sovellus on suunniteltu jaettavaksi neljään eri sovellustasoon. Lisäksi asiakkaan toivomuksena oli tuki erilaisten graafien piirtoon. Tätä tarkoitusta varten sovellukseen laaditaan erillinen rajapinta, jonka kautta erilaisten kuvaajien/kaavioiden piirtäminen on mahdollisimman helppoa.

## 3.3 Kokoarvio

OhtuTie-projektin tarkoituksena on laatia järjestelmä, jota voidaan helposti muokata ja johon voidaan nopeasti ja vaivattomasti lisätä uusia toimintoja. Tämänkaltaisen projektin toteuttaminen joustavasti vaatii metaohjelmoinnin (refleksiivisyyden) hyödyntämistä, mikä hankaloittaa kokoarvion tekemistä historiatiedon perusteella. Tämän takia kokoarvio perustuu pääasiallisesti ryhmän aiempaan kokemukseen.

Kokoarvio on tehty sovellustasoittain, koska tämä havainnollistaa parhaiten sovelluksen rakennetta sekä helpottaa toteutusvaiheen organisointia.

Asiakaskerros koostuu pääasiallisesti yhdestä Java Servlet komponentista, jonka tehtävänä on "ohjata" sivupyynnöt oikeille komponenteille. Servlet komponentin toteutus vaatii arviolta 500 riviä Java-koodia. Servletin toiminnot on hajautettu useisiin pienempiin komponentteihin, joista tärkein on LayoutManager- komponentti. LayoutManager- komponentin toteuttaminen vaatii arviolta 250 riviä Java-koodia. LayoutManager hyödyntää apukirjastoina toteutettavia XML- ja Metadata-kirjastoja. XML-kirjaston kokovaatimus on noin 500 riviä ja Metadata-kirjaston vaatimuksena on noin 250 koodiriviä. Jokainen sivu toteutetaan komponenttina, jolla on metadata-kuvaus sekä Java-luokka, joka toteuttaa määritellyn rajapinnan. Sivukomponentteja tarvittaneen arviolta 10 kappaletta, joista jokainen vaatii noin 200 koodiriviä (Java). Täten koko asiakaskerroksen kokovaatimukseksi saadaan 3500 riviä. Taulukossa 2 on eritelty asiakaskerroksen koodimäärä komponentteittain.

Taulukko 2: Asiakaskerroksen koodimäärä komponentteittain

<b>Komponentti</b>	<b>riviä</b>	<b>riviä</b>
Servlet		500
LayoutManager		250
XML apukirjasto		500
Metadata apukirjasto		250
Sivukomponentit	200 riviä * 10 ⇒	2000
=		<b>3500</b>

Näkymäkerroksen komponentit koostuvat pääasiallisesti sivupohjista (template) sekä niille määritellyistä XSL- muunnoksista (transformaatioista). Templateluokkien käsittelyyn tarvitaan erillinen TemplateManager Java -komponentti, jonka toteutus vaatii n. 250 riviä Java-koodia. Sivupohjien toteuttaminen vaatii noin 100 riviä/sivupohja (XML-kielellä). Muunnosten toteuttamiseen tarvitaan arviolta 250 riviä/muunnos. Muunnoksia tarvitaan sovelluksen tämän version toteuttamiseksi vain kaksi, joista ns. päämuunnospohjan toteutus vaatii n. 200 riviä XSL- koodia. Toisen, ns. sisältömuunnospohjan, toteutus vaatii noin 50 riviä XSL- koodia. Yhteenveto näkymäkerroksen komponenttien riviarviosta on esitetty taulukossa 3.

Taulukko 3: Yhteenveto näkymäkerroksen komponenttien riviarviosta

<b>Näkymäkomponentti</b>	<b>riviä</b>	<b>toteutuskieli</b>
TemplateManager	250	Java
Päämuunnospohja	200	XSL
Sisältömuunnospohja	50	XSL
Template	100 * 10 ⇒	1000
=		<b>1500</b>



Sovelluslogiikkakerros toteutaan Java- sekä XML-kielillä (metadata). Sovelluslogiikkakerros vastaa mm. sovelluksen tietoturvasta, tietokantaoperaatioista sekä istuntojen hallinnasta. Tarvittavat toiminnallisuudet toteutetaan komponentteina. SecurityManager-komponentti huolehtii sovelluksen tietoturvasta ja suorittaa salaus- ja purkuoperaatioita tarvittaessa. Tämän komponentin toteuttaminen ei ole välttämätöntä, mutta sovelluksen tietoturvan kannalta on suotavaa, jos tietoturvakomponentti toteutetaan. Komponentin rivi-arvio on 1000 riviä. Istunnonhallintakomponentin tehtävänä on mm. pitää kirjaa käyttäjän istunnosta (mm. kesto) sekä varmistaa, että käyttäjällä on riittävät oikeudet pyydettyihin komponentteihin. Tämän komponentin rivi-arvio on 1000 riviä Java-koodia. Viimeisenä sovelluslogiikkakerroksen komponenttina on tietokannanhallintakomponentti. Tämän komponentin tehtävänä on piilottaa yhteyden muodostamisrutiinit, helpottaa tietokantakyselyjen suorittamista sekä yleisesti eristää tietokanta muusta sovelluksesta. Komponentin toteuttaminen vaatii arviolta 1000 riviä Java-koodia. Taulukossa 4 on yhteenveto sovelluslogiikkakerroksen komponenttien rivi-arvioista.

Taulukko 4: Yhteenveto sovelluslogiikkakerroksen komponenttien rivi-arvioista

<b>Sovelluskerroksen komponentti</b>	<b>rivi-arvio</b>
SecurityManager	1000
SessionManager	1000
DatabaseManager	1000
=	<b>3000</b>

Itse tietokantaosuuden toteuttaminen ei käytettävässä 4-tier mallissa vaadi ylimääräistä koodaamista. Tietokannan eristämiseksi muusta sovelluksesta tarvitaan XML-muotoista metadataa. Kokonaisuudessaan metadataa vaaditaan jokaiselle komponentille, tietokannalle sekä eräiden rajapintojen (mm. template) omiin tarpeisiin. Metadataan määrää on hankala arvioida täsmällisesti saati erotella tarkemmin rivimäärää, joten esitämme vain karkean arvion **2000 riviä** tarvittavan metadataan määrälle. Metadataan määrä kasvaa/vähenee sen mukaan, mitä XML-tekniikkaa käytetään ja käytetäänkö DTD-, Schema- tai joitain muita muunnoksia.

Itse sovelluksen ulkopuolisena vaatimuksena on vielä rajapinnan määrittäminen erilaisien graafien toteuttamiseksi. Jotta joustava rajapinta olisi mahdollista toteuttaa, tarvitaan modulaarinen rakenne, jossa graafin osat on pilkottu hierarkkisesti pienempiin osiin. Tämnäkaltaisen rajapinnan toteuttaminen vaatii arviolta 2000 riviä Java-koodia. Taulukossa 5 on esitetty yhteenvetona sovelluksen kokoarvio luokiteltuna sovelluskerroksien mukaan.

Taulukko 5: Sovelluksen kokoarvio luokiteltuna sovelluskerroksien mukaan

<b>Vaihe</b>	<b>toteutuskielet</b>	<b>rivi-arvio</b>
Asiakaskerros	Java	3500
Näkymäkerros	XML,XSLT	1500
Sovelluslogiikkakerros	Java,XML	3000
Muu metadata	XML	2250
Graafikirjasto	Java	2000
=		<b>12000</b>

## 4 Projektin eteneminen

Luvussa kuvataan projektin etenemistä laaditun vaihejaon ja aikataulun mukaisesti. Projektin aikataulua muutetaan vain pakottavista syistä. Aikataulussa pysyminen edellyttää kultakin ryhmän jäseneltä vähintään 30 tuntia viikoittaista työaikaa siten, että vaadittava kokonaistyöaika 240 tuntia täyttyy projektin päättyessä.

### 4.1 Työvaiheet

Projekti toteutetaan käyttäen sovellettua lineaarista projektimallia, jonka osat ovat määrittely, suunnittelu, toteutus, testaus ja ylläpito. Suunnittelu-, toteutus- ja testausvaihe limityvät osittain edellisen vaiheen päälle, mutta kunkin vaiheen pääasiallinen toteutus alkaa suunnitellun aikataulun mukaisesti. Ohjelmiston ylläpito rajataan projektin ulkopuolelle. Sen sijaan asiakkaalle toimitetaan luovutusvaiheessa ylläpitodokumentti.

#### Määrittely

Määrittelyvaiheen alussa ryhmä laatii projektisuunnitelman. Se toimii projektiryhmän toimintaa ohjaavana dokumenttina projektin eri vaiheissa.

Määrittelyvaiheessa kartoitetaan ohjelmiston vaatimukset, päätoiminnot, liittymät ja rajoitukset. Kartoituksen tuloksen perusteella laaditaan asiakkaalle toimitettava määrittelydokumentti. Asiakas esittää dokumenttiin haluamansa muutokset ja korjaukset vaiheen lopussa järjestettävässä hyväksymiskokouksessa. Dokumentti päivitetään asiakkaan toiveita vastaavaksi suunnitteluvaiheen alussa.

Määrittelyvaiheen lopulla aloitetaan myös järjestelmän arkkitehtuurisuunnittelu.

#### Suunnittelu

Suunnitteluvaiheen aikana määritellään ohjelmiston lopullinen arkkitehtuuri: tietorakenteet, käyttöliittymä ja rajapinnat. Määrittelydokumentin perusteella laaditaan suunnitteludokumentti, joka tarkastetaan vaiheen päätteeksi hyväksymiskokouksessa. Esiin tulleet muutostarpeet päivitetään dokumenttiin viimeistään toteutusvaiheen alussa.

Suunnitteludokumentissa pyritään sellaiseen kuvaustarkkuuteen, että ohjelmointi voidaan tehdä suoraviivaisesti sen pohjalta.

Ohjelmointi- sekä testaussuunnitelman laadinta aloitetaan suunnitteluvaiheen aikana.

### **Toteutus**

Ohjelmointi tehdään pääasiassa toteutusvaiheessa ja se pyritään saattamaan loppuun koodin hyväksymiskokoukseen mennessä. Testaus aloitetaan toteutusvaiheessa ennen varsinaisen testausvaiheen alkua. Toteutusvaiheen aikana aloitetaan käyttöohjeen laadinta.

### **Testaus**

Varsinainen testaus saatetaan loppuun testausvaiheen aikana, jolloin ohjelmointi on jo päättynyt. Testausmenetelmät ja -tulokset dokumentoidaan testausdokumentissa. Testausvaiheen lopuksi valmistuu toteutusdokumentti, käyttöohje sekä ylläpitodokumentti.

### **Projektin päättäminen**

Projekti päättyy 9.7.2004. Projekti katsotaan päättyneeksi, kun asiakkaalle on luovutettu tilattu ohjelmisto lähdekoodeineen, ohjelman käyttöohje sekä loppuraportti. Luovutettavat dokumentit ja ohjelmisto toimitetaan sähköisessä muodossa CD-ROM -levykkeellä. Projektin palautetilaisuus järjestetään oppimiskeskus Aleksandrian ryhmätyötiloissa maanantaina 26.7.2004. Palautetilaisuudessa ohjelmistotuotantoprojektin ohjaaja, vastuuhenkilö ja asiakas arvioivat projektin onnistumista sekä annettua arvosanaa.

Jotta projektin päättäminen on mahdollista aikataulun mukaisesti, on kaikkien työvaiheiden oltava valmiina keskiviikkona 7.7.2004 työpäivän päättyessä. Torstai 8.7.2004 käytetään viimeisiin tarkastuksiin ja vain erityistarpeessa viimeistelytyöhön.

Projekti päätetään demotilaisuudessa perjantaina 9.7.2004. Materiaali luovutetaan tilaisuuden päätteeksi asiakkaalle tai hänen edustajalleen.

## **4.2 Aikataulu**

Projektin aikataulu on esitetty taulukossa 6 sekä liitteenä olevassa Gantt-kaaviossa.

## **5 Työskentelytavat ja viestintä**

Luvussa kuvataan projektin kokousmenettelyt, päivittäiset työrutiinit, työskentelypaikat ja ryhmän sisäinen viestintä.

### **5.1 Kokoukset**

Projektiryhmä aloittaa päivittäisen työskentelynsä ryhmäpalaverilla klo 09.00 alkaen. Palaverit pyritään pitämään mahdollisimman lyhyinä ja ytimekkäinä. Jos ryhmän jäsen ei pääse palaveriin perustellusta syystä, hän ilmoittaa siitä mahdollisimman aikaisin ohjaajalle tai projektipäällikölle joko suullisesti tai puhelimitse. Myös sähköpostilistaa voi käyttää, mutta se on nopeissa tilanteissa toissijainen menettely. Jos palaveri katsotaan erityisen tärkeäksi, se voidaan siirtää poissaolon vuoksi samalle päivälle erikseen sovittavaan ajankohtaan.

Palaveri on yleiskokous, jossa käsitellään ja päätetään työntekoon olennaisesti vaikuttavat yhteiset asiat ja määritellään päivittäinen työnjako projektin jäsenten kesken. Henkilökoh-  
taisissa projektin jäsenten välisissä keskusteluissa ei tehdä projektiin liittyviä linjanvetoja,  
vaan laajemmista työtä ohjaavista menettelyistä päätetään palavereissa.

Projektin ohjaaja on läsnä aamupalavereissa ja projektin työskentelyssä tarpeelliseksi kat-  
somansa ajan. Tarkastuskokouksen puheenjohtajana toimii ohjaaja. Muita läsnäolijoita  
kokouksissa ovat tarvittaessa asiakas ja projektin vastuuhenkilö.

Hyväksymis- ja tarkastuskokouksiin osallistuville toimitetaan käsiteltävät dokumentit säh-  
köpostitse pdf-muotoisena liitetiedostona.

### **5.1.1 Seurantakokoukset**

Seurantakokoukset pidetään viikon päätteeksi aamupalaverin yhteydessä. Kokouksen kes-  
to on korkeintaan puoli tuntia. Projektipäällikkö laatii kokoukselle esityslistan, jonka hän  
toimittaa edellisenä päivänä ryhmän jäsenille ryhmän sähköpostilistan kautta. Ensimmäi-  
nen seurantakokous pidetään 28.05.04.

Kokouksessa käydään läpi työvaiheen ja koko projektin edistymistä, arvioidaan aikatau-  
lun pitämistä ja käydään läpi mahdolliset toteutuneet riskit. Seurantakokouksen perus-  
teella projektipäällikkö kirjaa mahdollisesti toteutuneet muutokset projektisuunnitelmaan.  
Seurantakokouksessa päätetään myös muihin hyväksytyihin työvaiheen päättäviin doku-  
mentteihin tehtävistä muutoksista.

### **5.1.2 Hyväksymiskokoukset**

Hyväksymiskokous pidetään aina työvaiheen päätteeksi ja siinä tarkastetaan vaiheen tuot-  
tama dokumentti. Toteutusvaiheen päätteeksi pidetään myös hyväksymiskokous, jossa  
käydään läpi vaiheen tuottamaa ohjelmakoodia.

### **5.1.3 Tarkastuskokous**

Tarkastuskokouksessa käydään läpi suunnitteludokumentti. Kokoukseen osallistuvat saa-  
vat dokumentin tutustumista varten yhtä työpäivää aikaisemmin. Dokumentissa havaitut  
virheet tai puutteet kirjataan kokouksen aikana ja ne korjataan myöhemmin erikseen mää-  
ritettävän henkilön toimesta. Korjaukset hyväksytetään puheenjohtajalla. Hyväksymisen  
jälkeen suunnitteludokumentti jäädytetään ja dokumenttiin tehtävät muutokset päätetään  
tämän jälkeen seurantakokouksessa.

## **5.2 Työskentelyajat ja -paikat**

Projektiryhmän päivittäinen oletustyöaika on kuusi tuntia projektin kahdeksan viikon kes-  
ton ajan siten, että kunkin jäsenen 240 tunnin kokonaistyöaika vaatimus tulee täyteen.

Edellytyksenä on kuitenkin vähintään 30 tunnin viikoittainen työaika täysille viiden päivän työviikoille laskettuna. Juhannusaatto 25.6.04 ei ole työpäivä normaalin työaikamennettelyn mukaisesti.

Koska projekti toteutetaan nopealla aikataululla, ryhmä on päättänyt työskennellä yhdessä samoissa tiloissa. Tällä pyritään sisäisen kommunikaation helpottamiseen ja siten ongelmatilanteiden mahdollisimman nopeaan ratkaisuun. Halutessaan projektin jäsen voi työskennellä esimerkiksi kotonaan, mutta se ei poista osallistumisvelvollisuutta aamupalaveriin ja muihin erikseen määriteltyihin tilaisuuksiin.

Juhannuksen jälkeen 28.6.04 alkaen aamupalaverit ja ryhmän yhteinen työskentely siirretään Helsingin yliopiston oppimiskeskus Aleksandriaan, jos TKTL:n opiskelijoille tarkoitettut tietokoneet poistuvat käytöstä ennen projektin päättymistä. Projektipäällikkö huolehtii tällöin ryhmän työskentelyyn tarvittavien ryhmätyötilojen varaamisesta Aleksandriasesta.

## 6 Dokumentointi ja raportointi

Luvussa kuvataan projektin tuottamat dokumentit ja niiden yleissisällöt. Lisäksi kuvataan dokumentaatioon liittyviä käytännön menettelyjä. Projektissa käytettävän sovelletun lineaarisen mallin kunkin työvaiheen tulodokumentti toimii syötteenä seuraavalle vaiheelle.

### 6.1 Yleismenettelyt

Ryhmän tuottamat dokumentit, lähdekoodi sekä varsinainen tuotettu ohjelmisto linkitetään ryhmän kotisivuille. Ryhmän kotihakemistoon `/home/group/otie` luodaan viikoittain uusi viikkohakemisto, jonne kopioidaan työviikon päätteeksi kaikki viikon aikana tuotettu materiaali.

Taulukossa 7 esitetään projektissa valmistuvat dokumentit vaiheittain. Vaatimusmäärittelydokumentti tarkastetaan yhdessä asiakkaan kanssa hyväksymiskokouksessa. Muut dokumentit tarkastetaan ryhmän sisäisesti hyväksymiskokouksessa. Projektin dokumentointi- ja raportointikielenä on suomi.

### 6.2 Tuotettavat dokumentit

#### Projektisuunnitelma

Projektisuunnitelma sisältää projektin osapuolet, projektin aikataulun ja tuotettavat dokumentit. Projektisuunnitelma pitää projektin aikataulussa

#### Määrittelydokumentti

Hyväksymismenettely: Asiakas hyväksyy hyväksymiskokouksessa.

Määrittelydokumentissa kuvataan vaatimusanalyysin perusteella määritellyt tuotteelta vaaditut ominaisuudet.

**Suunnitteludokumentti**

Hyväksymismenettely: Hyväksymiskokous

Suunnitteludokumentti kuvaa tuotettavan ohjelmiston rakenteen ja toiminnan. Dokumentti sisältää tieto-, arkkitehtuuri-, liittymä- ja moduulisuunnitelman.

**Toteutusdokumentti**

Toteutusdokumentti tai päivitetty suunnitteludokumentti kertoo miten toteutus onnistui.

**Testaussuunnitelma**

Sisältää kuvauksen tehtävistä moduuli-, luokka- ja integrointitason testeistä.

**Testausdokumentti**

Sisältää testaussuunnitelman mukaisesti tehdyt testiaineistot ja testien tulokset.

**Käyttöohje**

Hyväksymismenettely: Asiakas hyväksyy hyväksymiskokouksessa.

Sisältää loppukäyttäjälle tarkoitettun asennus- ja käyttöohjeen sekä kuvauksen järjestelmän toiminnoista.

**Ylläpidodokumentti**

Sisältää kuvauksen ohjelmiston toimintaperiaatteista ylläpidon avuksi.

**Loppuraportti**

Sisältää projektityhteenvedon. Tähän dokumenttiin liitetään projektin sisäiset dokumentit: työtuntilistat, pöytäkirjat ja sisäiset muistiot.

## 7 Ohjaus ja seuranta

### 7.1 Edistymisen seuranta

Projektin edistymistä seurataan päivittäin projektiryhmän kokouksissa, joissa on läsnä ryhmän jäsenten lisäksi myös projektin ohjaaja.

Viikkotasolla projektin edistymistä seurataan projektin vastuuhenkilölle ja projektiryhmälle toimitettavissa edistymisraporteissa, jotka projektipäällikkö koostaa ja lähettää edelleen maanantaisin. Edistymisraporttiin kootaan tiedot edellisen viikon työtunneista sekä viikon aikana valmistuneista dokumenteista. Lisäksi edistymistä seurataan perjantaiamuksin pidettävissä sisäisissä seurantakokouksissa.

### 7.2 Projektisuunnitelman ylläpito

Projektipäällikkö ylläpitää projektisuunnitelmaa tarvittaessa, kuitenkin vähintään kerran viikossa seurantakokouksen jälkeen.

### **7.3 Tulosten hyväksyminen**

Projektin vastuuhenkilö hyväksyy projektin väli- ja lopputulokset. Hyväksyntää varten valmistellaan kustakin vaiheesta siihen liittyvä dokumentti, joka käsitellään hyväksymiskokouksessa. Suunnitteludokumentti käsitellään tarkastuskokouksessa.

### **7.4 Muutosten, virheiden ja ongelmien hallinta**

Muutosten hallinta on oleellinen osa projektityöstekentelyä. Jäädetyttyä dokumenttia voidaan muuttaa vain erityisen muutosprosessin kautta. Projektin vastuuhenkilö tekee lopullisen päätöksen muutosten hyväksymisestä tai hylkäämisestä.

Virheitä ja ongelmia hallitaan pääasiassa seuranta- ja hyväksymiskokousten avulla. Hyväksymiskokouksissa havaitut virheet, puutteet ja ongelmat merkitään kokouspöytäkirjaan ja osoitetaan jonkun henkilön korjattavaksi. Korjaukset hyväksytetään kokouksen puheenjohtajalla.

### **7.5 Laadun valvonta**

Projektiryhmän laadunvalvonta käsittää prosessin ja tuotteen laadun valvonnan.

Prosessin laadunvalvonta tapahtuu pitäytymällä mahdollisimman tarkoin prosessimallin määrittämässä työvaiheissa ja noudattamalla projektin aikataulua. Prosessin laatua valvotaan myös tarkastamalla seuraavan vaiheen syötedokumentti hyväksymiskokouksissa ja tarkastuskokouksessa.

Dokumentin laatu käsittää mm. dokumentin kattavuuden, sisällön luettavuuden ja ulkoasun selkeyden.

Tuotteen laatuun vaikuttaa olennaisesti määrittelyn kattavuus. Suunnitteludokumentin osuus on myös erittäin tärkeä, koska ohjelmointi tehdään sen pohjalta. Tuotteen eli ohjelmiston laadun valvontaa suoritetaan toteutusvaiheen päätteeksi järjestetävässä ohjelmakoodin hyväksymiskokouksessa, jossa käydään läpi ohjelmakoodin kommentointi ja yleisrakenne. Lisäksi kattavalla testauksella varmistetaan ohjelman virheetön ja oikea toiminta.

Tuotteen eli ohjelmiston laadulla käsitetään toteutettujen määrittelyjen kattavuutta sekä ohjelmiston oikeaa ja luotettavaa toimintaa.

## **8 Riskianalyysi**

Luvussa kuvataan projektin toteutukseen liittyvät riskit ja niiden hallintakeinot. Riskien todennäköisyyttä ja vakavuutta arvioidaan asteikolla pieni, kohtalainen, suuri.

## 8.1 Eriävät näkemykset toteutettavasta järjestelmästä

Mahdolliset asiakkaan, ohjaajan ja projektiryhmän jäsenten toisistaan eriävät näkemykset järjestelmästä tuovat mukanaan omat riskinsä. Aikataulun tiukkuudesta ja lasketun työmäärän (240 h \* 6) rajallisuudesta johtuen kaikkia ideoita ja vaatimuksia ei pystytä toteuttamaan projektille annetun ajan puitteissa.

## 8.2 Projektiryhmään liittyvät riskit

**Riski:** Ryhmän jäsen keskeyttää projektin.

**Vastatoimet:** Hyvän työilmapiirin ylläpito. Projektin keskeyttävä ilmoittaa aikeesta ajoissa ryhmälle. Pyritään siihen ettei mikään osa-alue ole vain yhden ryhmän jäsenen hoidettavana.

**Todennäköisyys:** Pieni.

**Vakavuus:** Suuri.

**Riski:** Ryhmän jäsen sairastuu tai on muuten poissa.

**Vastatoimet:** Hyvän työilmapiirin ylläpito. Muut hoitavat poissaolevan osuutta.

**Todennäköisyys:** Pieni.

**Vakavuus:** Kohtalainen.

## 8.3 Projektin hallintaan liittyvät riskit

**Riski:** Arviointiriski (työmäärä / osaamistaso).

**Vastatoimet:** Päivitetään aikataulua.

**Todennäköisyys:** Suuri.

**Vakavuus:** Kohtalainen.

**Riski:** Laitoksen muutosta aiheutuvat ongelmat. Projektille ei löydy sopivia työtiloja.

**Vastatoimet:** Etupainotteinen työskentely, jolloin projekti valmistuu viimeistään aikataulun mukaisesti. Selvitetään muita mahdollisia työtiloja: Aleksandria, Kumpula (fysiikan laitos), kotona.

**Todennäköisyys:** Pieni.

**Vakavuus:** Kohtalainen.

## 8.4 Tekniikkaan liittyvät riskit

**Riski:** Valitut ohjelmat osoittautuvat vaikeiksi käyttää tai oppia (UML-työkalu, kehitysympäristö, projektinhallintaohjelma, latex ym.).

**Vastatoimet:** Kokeillaan tarvittavia ohjelmia etukäteen.



**Todennäköisyys:** Suuri.

**Vakavuus:** Kohtalainen.

**Riski:** Tietokoneiden ja ohjelmistojen saatavuudesta aiheutuvat ongelmat (työasemat ja palvelimet).

**Vastatoimet:** Omien tietokoneiden käyttö. Tarkistetaan etukäteen mitä esimerkiksi Aleksandriassa on tarjolla.

**Todennäköisyys:** Suuri.

**Vakavuus:** Kohtalainen.

## 8.5 Tuotteeseen ja asiakkaaseen liittyvät riskit

**Riski:** Asiakas ei ole tavoitettavissa.

**Vastatoimet:** Tehdään määrittely huolellisesti yhteistyössä asiakkaan kanssa, jotta asiakasta ei tarvita enää projektin loppupuolella niin paljon. Jos asiakas ei ole tavoitettavissa, toimitaan olemassaolevin tiedoin.

**Todennäköisyys:** Pieni.

**Vakavuus:** Suuri.

**Riski:** Vaatimukset muuttuvat määrittelyvaiheen jälkeen.

**Vastatoimet:** Tehdään ohjelmistosta helposti muutettava. Kohtuuttomia muutoksia ei toteuteta.

**Todennäköisyys:** Suuri.

**Vakavuus:** Kohtalainen.

Taulukko 6: Projektiryhmän aikataulu

Vaihe	Mitä tehdään	Pvm
Määrittely	Projektisi alkaa	17.5.2004
	Projektisuunnitelman laatiminen alkaa	17.5.2004
	Määrittelydokumentin laatiminen alkaa	24.5.2004
	Seurantakokous	28.5.2004
	Määrittelydokumentin toimitus asiakkaalle	1.6.2004
	Määrittelydokumentin hyväksymiskokous	3.6.2004
Suunnittelu	Suunnitteludokumentin laatiminen alkaa	31.5.2004
	Seurantakokous	4.6.2004
	Testaussuunnitelman laatiminen alkaa	7.6.2004
	Seurantakokous	11.6.2004
	Suunnitteludokumentti valmis	23.6.2004
	Suunnitteludokumentin tarkastuskokous	24.6.2004
Toteutus	Ohjelmointi alkaa	14.6.2004
	Seurantakokous	18.6.2004
	Toteutusdokumentin laatiminen alkaa	24.6.2004
	Seurantakokous	24.6.2004
	Käyttöohjeen laatiminen alkaa	2.7.2004
Testaus	Testaus alkaa	28.6.2004
	Loppuraportin laatiminen alkaa	2.7.2004
	Testaussuunnitelma valmis	2.7.2004
	Käyttöohjeen esiversio asiakkaalle	6.7.2004
	Toteutusdokumentin esiversio suunnitteludokumentin pohjalta	6.7.2004
	Ylläpidodokumentin laatiminen alkaa	6.7.2004
	Seurantakokous	8.7.2004
	Testaus päättyy	8.7.2004
	Käyttöohje valmis	9.7.2004
	Toteutusdokumentti valmis	9.7.2004
	Ylläpidodokumentti valmis	9.7.2004
	Loppuraportti valmis	9.7.2004
Luovutus	Demotilaisuus asiakkaalle	9.7.2004
	Ohjelmiston, käyttöohjeen ja dokumentaation luovutus asiakkaalle	9.7.2004
	Projektisi päättyy	9.7.2004
Palaute	Palautetilaisuus Aleksandriassa	26.7.2004

Taulukko 7: Dokumentit ja niiden valmistumisaikataulut

Vaihe	Dokumentti	Valmistuu	Huomautuksia
Määrittely	Projektisuunnitelma	pe 28.05.04	
	Määrittelydokumentti	ti 01.06.04	Asiakas hyväksyy
Suunnittelu	Suunnitteludokumentti	to 10.06.04	Käsitellään tarkastuskokouksessa
	Testaussuunnitelma	to 10.06.04	
Toteutus	Toteutusdokumentin esiversio	to 24.06.04	
Testaus	Käyttöohjeen esiversio	ti 06.07.04	Asiakkaan arvioitavaksi
	Käyttöohje	pe 09.07.04	
	Testausdokumentti	pe 09.07.04	
	Toteutusdokumentti	pe 09.07.04	
	Ylläpitodokumentti	pe 09.07.04	
	Loppuraportti	pe 09.07.04	